



Consolidated Platform Configuration Guide, Cisco IOS Release 15.2(7)Ex (Catalyst 3560-CX and 2960-CX Switches)

First Published: 2019-03-27

Last Modified: 2022-09-29

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2020 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface	lxxxiii
Document Conventions	lxxxiii
Related Documentation	lxxxv
Obtaining Documentation and Submitting a Service Request	lxxxv

PART I

Interface and Hardware	87
-------------------------------	-----------

CHAPTER 1

Configuring Interface Characteristics	1
Information About Configuring Interface Characteristics	1
Interface Types	1
Port-Based VLANs	1
Switch Ports	2
Switch Virtual Interfaces	2
EtherChannel Port Groups	3
Power over Ethernet Ports	4
Using the Switch USB Ports	4
USB Mini-Type B Console Port	4
USB Type A Ports	5
Interface Connections	5
Interface Configuration Mode	6
Default Ethernet Interface Configuration	6
Interface Speed and Duplex Mode	7
Speed and Duplex Configuration Guidelines	7
IEEE 802.3x Flow Control	8
How to Configure Interface Characteristics	9
Configuring Interfaces	9

- Adding a Description for an Interface 10
- Configuring a Range of Interfaces 11
- Configuring and Using Interface Range Macros 12
- Configuring Ethernet Interfaces 14
 - Setting the Interface Speed and Duplex Parameters 14
- Configuring IEEE 802.3x Flow Control 15
- Configuring SVI Autostate Exclude 16
- Shutting Down and Restarting the Interface 17
- Configuring the Console Media Type 19
- Configuring the USB Inactivity Timeout 20
- Monitoring Interface Characteristics 21
 - Monitoring Interface Status 21
 - Clearing and Resetting Interfaces and Counters 22
- Configuration Examples for Interface Characteristics 22
 - Configuring a Range of Interfaces: Examples 22
 - Configuring and Using Interface Range Macros: Examples 23
 - Setting Interface Speed and Duplex Mode: Example 23
 - Configuring the Console Media Type: Example 23
 - Configuring the USB Inactivity Timeout: Example 24

CHAPTER 2

- Configuring Auto-MDIX 25**
 - Prerequisites for Auto-MDIX 25
 - Restrictions for Auto-MDIX 25
 - Information About Configuring Auto-MDIX 25
 - Auto-MDIX on an Interface 25
 - How to Configure Auto-MDIX 26
 - Configuring Auto-MDIX on an Interface 26
 - Example for Configuring Auto-MDIX 27

CHAPTER 3

- Configuring LLDP, LLDP-MED, and Wired Location Service 29**
 - Information About LLDP, LLDP-MED, and Wired Location Service 29
 - LLDP 29
 - LLDP Supported TLVs 29
 - LLDP and Cisco Medianet 30

LLDP-MED	30
LLDP-MED Supported TLVs	30
Wired Location Service	31
Default LLDP Configuration	32
Restrictions for LLDP	33
How to Configure LLDP, LLDP-MED, and Wired Location Service	33
Enabling LLDP	33
Configuring LLDP Characteristics	35
Configuring LLDP-MED TLVs	37
Configuring Network-Policy TLV	38
Configuring Location TLV and Wired Location Service	40
Enabling Wired Location Service on the Device	43
Configuration Examples for LLDP, LLDP-MED, and Wired Location Service	44
Configuring Network-Policy TLV: Examples	44
Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service	45

CHAPTER 4 **Configuring MultiGigabit Ports on WS-C3560CX-8PD-S** 47

Finding Feature Information	47
Overview of MultiGigabit Ports	47
Restrictions for MultiGigabit Ports	48
Supported Cable Types and Maximum Length	48
Setting the Interface Speed	48
Examples: Setting the Interface Speed	49

CHAPTER 5 **Configuring System MTU** 51

Information About the MTU	51
System MTU Guidelines	51
How to Configure MTU	51
Configuring the System MTU	51
Configuration Examples for System MTU	52

CHAPTER 6 **Configuring Boot Fast** 53

Configuring Boot Fast on the switch	53
Enabling Boot Fast	53

Disabling Boot Fast 54

CHAPTER 7

Configuring Power over Ethernet 57

Information About PoE 57

Power over Ethernet Ports 57

PoE and PoE Pass-Through Ports on Catalyst WS-C3560CX-8PT-S 57

Example: Configuring PoE and PoE Pass-Through Ports on WS-C3560CX-8PT-S 58

Supported Protocols and Standards 59

Powered-Device Detection and Initial Power Allocation 59

Power Management Modes 60

How to Configure PoE 63

Configuring a Power Management Mode on a PoE Port 63

Configuring PoE and PoE Pass-Through Ports on Catalyst WS-C3560CX-8PT-S 65

Perpetual POE 65

Fast POE 65

Configuring Persistent and Fast PoE 66

Budgeting Power for Devices Connected to a PoE Port 67

Budgeting Power to All PoE ports 67

Budgeting Power to a Specific PoE Port 68

Configuring Power Policing 69

Monitoring Power Status 72

Configuration Examples for Configuring PoE 72

Budgeting Power: Example 72

CHAPTER 8

Configuring 2-event Classification 73

Information about 2-event Classification 73

Configuring 2-event Classification 73

Example: Configuring 2-Event Classification 74

CHAPTER 9

Configuring EEE 75

Restrictions for EEE 75

Information About EEE 75

EEE Overview 75

Default EEE Configuration 75

How to Configure EEE	75
Enabling or Disabling EEE	76
Monitoring EEE	77
Configuration Examples for Configuring EEE	77

PART II
IP Multicast Routing 79

CHAPTER 10
IP Multicast Routing Technology Overview 81

Information About IP Multicast Technology	81
Role of IP Multicast in Information Delivery	81
IP Multicast Routing Protocols	81
Multicast Group Transmission Scheme	81
IP Multicast Boundary	82
IP Multicast Group Addressing	83
IP Class D Addresses	83
IP Multicast Address Scoping	83
Layer 2 Multicast Addresses	85
IP Multicast Delivery Modes	85
Source Specific Multicast	85

CHAPTER 11
Configuring Basic IP Multicast Routing 87

Prerequisites for Basic IP Multicast Routing	87
Restrictions for Basic IP Multicast Routing	87
Information About Basic IP Multicast Routing	87
Default IP Multicast Routing Configuration	88
sdr Listener Support	88
How to Configure Basic IP Multicast Routing	89
Configuring Basic IP Multicast Routing	89
Configuring Optional IP Multicast Routing Features	91
Defining the IP Multicast Boundary	91
Configuring Multicast VRFs	92
Advertising Multicast Multimedia Sessions Using SAP Listener	94
Monitoring and Maintaining Basic IP Multicast Routing	96
Clearing Caches, Tables, and Databases	96

Displaying System and Network Statistics 96

CHAPTER 12

Configuring IGMP 99

Prerequisites for IGMP 99

Restrictions for Configuring IGMP 99

Information About IGMP 100

Role of the Internet Group Management Protocol 100

IGMP Multicast Addresses 100

IGMP Versions 100

IGMP Version 1 101

IGMP Version 2 101

IGMP Version 3 101

IGMPv3 Host Signalling 101

IGMP Versions Differences 101

IGMP Join and Leave Process 103

IGMP Join Process 103

IGMP Leave Process 104

Default IGMP Configuration 104

How to Configure IGMP 105

Configuring the Device as a Member of a Group 105

Controlling Access to IP Multicast Group 106

Changing the IGMP Version 108

Modifying the IGMP Host-Query Message Interval 110

Changing the IGMP Query Timeout for IGMPv2 111

Changing the Maximum Query Response Time for IGMPv2 113

Configuring the Device as a Statically Connected Member 114

Monitoring IGMP 115

Configuration Examples for IGMP 116

Example: Configuring the Device as a Member of a Multicast Group 116

Example: Controlling Access to IP Multicast Groups 116

CHAPTER 13

Configuring IGMP Snooping and Multicast VLAN Registration 117

Prerequisites for Configuring IGMP Snooping and MVR 117

Prerequisites for IGMP Snooping 117

Prerequisites for MVR	118
Restrictions for Configuring IGMP Snooping and MVR	118
Restrictions for IGMP Snooping	118
Restrictions for MVR	118
Information About IGMP Snooping and MVR	119
IGMP Snooping	119
IGMP Versions	120
Joining a Multicast Group	120
Leaving a Multicast Group	122
Immediate Leave	123
IGMP Configurable-Leave Timer	123
IGMP Report Suppression	123
Default IGMP Snooping Configuration	124
Multicast VLAN Registration	124
MVR and IGMP	124
Modes of Operation	125
MVR in a Multicast Television Application	125
Default MVR Configuration	127
IGMP Filtering and Throttling	127
Default IGMP Filtering and Throttling Configuration	128
How to Configure IGMP Snooping and MVR	128
Enabling or Disabling IGMP Snooping on a Device	128
Enabling or Disabling IGMP Snooping on a VLAN Interface	129
Setting the Snooping Method	130
Configuring a Multicast Router Port	132
Configuring a Host Statically to Join a Group	133
Enabling IGMP Immediate Leave	135
Configuring the IGMP Leave Timer	136
Configuring TCN-Related Commands	137
Controlling the Multicast Flooding Time After a TCN Event	137
Recovering from Flood Mode	139
Disabling Multicast Flooding During a TCN Event	140
Configuring the IGMP Snooping Querier	141
Disabling IGMP Report Suppression	143

- Configuring MVR Global Parameters 144
- Configuring MVR Interfaces 147
- Configuring IGMP Profiles 149
- Applying IGMP Profiles 151
- Setting the Maximum Number of IGMP Groups 153
- Configuring the IGMP Throttling Action 154
- Monitoring IGMP Snooping and MVR 156
 - Monitoring IGMP Snooping Information 156
 - Monitoring MVR 157
 - Monitoring IGMP Filtering and Throttling Configuration 158
- Configuration Examples for IGMP Snooping and MVR 158
 - Example: Configuring IGMP Snooping Using CGMP Packets 158
 - Example: Enabling a Static Connection to a Multicast Router 159
 - Example: Configuring a Host Statically to Join a Group 159
 - Example: Enabling IGMP Immediate Leave 159
 - Example: Setting the IGMP Snooping Querier Source Address 159
 - Example: Setting the IGMP Snooping Querier Maximum Response Time 159
 - Example: Setting the IGMP Snooping Querier Timeout 159
 - Example: Setting the IGMP Snooping Querier Feature 160
 - Example: Configuring IGMP Profiles 160
 - Example: Applying IGMP Profile 160
 - Example: Setting the Maximum Number of IGMP Groups 160
 - Example: Configuring MVR Global Parameters 160
 - Example: Configuring MVR Interfaces 161

CHAPTER 14

Configuring CGMP 163

- Finding Feature Information 163
- Prerequisites for Configuring CGMP 163
- Restrictions for CGMP 163
- Information About CGMP 164
- Enabling CGMP Server Support 164
- Monitoring CGMP 166

CHAPTER 15

Configuring Protocol Independent Multicast (PIM) 169

Prerequisites for PIM	169
Restrictions for PIM	170
PIMv1 and PIMv2 Interoperability	170
Restrictions for Configuring PIM Stub Routing	170
Restrictions for Configuring Auto-RP and BSR	171
Information About PIM	172
Protocol Independent Multicast	172
PIM Dense Mode	172
PIM Sparse Mode	173
Sparse-Dense Mode	173
PIM Versions	174
PIM Stub Routing	174
IGMP Helper	175
Rendezvous Points	176
Auto-RP	176
Sparse-Dense Mode for Auto-RP	177
Bootstrap Router	177
PIM Domain Border	178
Multicast Forwarding	178
Multicast Distribution Source Tree	178
Multicast Distribution Shared Tree	179
Source Tree Advantage	180
Shared Tree Advantage	180
PIM Shared Tree and Source Tree	181
Reverse Path Forwarding	183
RPF Check	184
Default PIM Routing Configuration	185
How to Configure PIM	185
Enabling PIM Stub Routing	185
Configuring a Rendezvous Point	187
Manually Assigning an RP to Multicast Groups	187
Setting Up Auto-RP in a New Internetwork	189
Adding Auto-RP to an Existing Sparse-Mode Cloud	192
Configuring Sparse Mode with a Single Static RP(CLI)	195

Preventing Join Messages to False RPs	197
Filtering Incoming RP Announcement Messages	197
Configuring PIMv2 BSR	199
Defining the PIM Domain Border	199
Defining the IP Multicast Boundary	201
Configuring Candidate BSRs	202
Configuring the Candidate RPs	204
Delaying the Use of PIM Shortest-Path Tree	206
Modifying the PIM Router-Query Message Interval	208
Verifying PIM Operations	209
Verifying IP Multicast Operation in a PIM-SM or a PIM-SSM Network	209
Using PIM-Enabled Routers to Test IP Multicast Reachability	215
Monitoring and Troubleshooting PIM	217
Monitoring PIM Information	217
Monitoring the RP Mapping and BSR Information	217
Troubleshooting PIMv1 and PIMv2 Interoperability Problems	217
Configuration Examples for PIM	218
Example: Enabling PIM Stub Routing	218
Example: Verifying PIM Stub Routing	218
Example: Manually Assigning an RP to Multicast Groups	218
Example: Configuring Auto-RP	219
Example: Defining the IP Multicast Boundary to Deny Auto-RP Information	219
Example: Filtering Incoming RP Announcement Messages	219
Example: Preventing Join Messages to False RPs	219
Example: Configuring Candidate BSRs	220
Example: Configuring Candidate RPs	220
CHAPTER 16	Configuring HSRP Aware PIM
	221
HSRP Aware PIM	221
Restrictions for HSRP Aware PIM	221
Information About HSRP Aware PIM	221
HSRP	221
HSRP Aware PIM	222
How to Configure HSRP Aware PIM	223

Configuring an HSRP Group on an Interface	223
Configuring PIM Redundancy	225
Configuration Examples for HSRP Aware PIM	226
Example: Configuring an HSRP Group on an Interface	226
Example: Configuring PIM Redundancy	226

CHAPTER 17**Configuring VRRP Aware PIM 227**

VRRP Aware PIM	227
Restrictions for VRRP Aware PIM	227
Information About VRRP Aware PIM	227
Overview of VRRP Aware PIM	227
How to Configure VRRP Aware PIM	228
Configuring VRRP Aware PIM	228
Configuration Examples for VRRP Aware PIM	230
Example: VRRP Aware PIM	230

CHAPTER 18**Configuring SSM 231**

Prerequisites for Configuring SSM	231
Restrictions for Configuring SSM	231
Information About SSM and SSM Mapping	233
SSM Components	233
How SSM Differs from Internet Standard Multicast	233
SSM Operations	234
IGMPv3 Host Signaling	234
Benefits of	234
SSM Mapping Overview	236
Static SSM Mapping	236
DNS-Based SSM Mapping	236
SSM Mapping Benefits	238
How to Configure SSM and SSM Mapping	238
Configuring SSM	238
Configuring SSM Mapping	240
Configuring Static SSM Mapping	240
Configuring DNS-Based SSM Mapping (CLI)	241

Configuring Static Traffic Forwarding with SSM Mapping	243
Verifying SSM Mapping Configuration and Operation	244
Monitoring SSM and SSM Mapping	246
Monitoring SSM	246
Monitoring SSM Mapping	247
Configuration Examples for SSM and SSM Mapping	247
SSM with IGMPv3 Example	247
SSM Filtering Example	248
SSM Mapping Example	248
DNS Server Configuration Example	251

CHAPTER 19**Configuring MSDP 253**

Prerequisites for MSDP	253
Information About Multicast Source Discovery Protocol	253
253	
MSDP Benefits	255
Default MSDP Peers	255
MSDP Mesh Groups	256
Benefits of MSDP Mesh Groups	257
SA Origination Filters	257
Use of Outgoing Filter Lists in MSDP	257
Use of Incoming Filter Lists in MSDP	258
TTL Thresholds in MSDP	259
MSDP Message Types	259
SA Messages	259
SA Request Messages	260
SA Response Messages	260
Keepalive Messages	260
Default MSDP Configuration	260
How to Configure MSDP	260
Configuring a Default MSDP Peer	260
Caching Source-Active State	262
Requesting Source Information from an MSDP Peer	263
Controlling Source Information that Your Switch Originates	264

Redistributing Sources	265
Filtering Source-Active Request Messages	267
Controlling Source Information that Your Switch Forwards	268
Using a Filter	268
Using TTL to Limit the Multicast Data Sent in SA Messages	270
Controlling Source Information that Your Switch Receives	271
Configuring an MSDP Mesh Group	273
Shutting Down an MSDP Peer	275
Including a Bordering PIM Dense-Mode Region in MSDP	276
Configuring an Originating Address other than the RP Address	277
Monitoring and Maintaining MSDP	278
Monitoring MSDP	278
Clearing MSDP Connections Statistics and SA Cache Entries	281
Configuration Examples for Configuring MSDP	282
Configuring a Default MSDP Peer: Example	282
Caching Source-Active State: Example	282
Requesting Source Information from an MSDP Peer: Example	282
Controlling Source Information that Your Switch Originates: Example	283
Controlling Source Information that Your Switch Forwards: Example	283
Controlling Source Information that Your Switch Receives: Example	283
Example: Configuring MSDP Mesh Groups	283
Requesting Source Information from an MSDP Peer: Example	284

PART III
IPv6 285

CHAPTER 20
Configuring MLD Snooping 287

Finding Feature Information	287
Information About Configuring IPv6 MLD Snooping	287
Understanding MLD Snooping	287
MLD Messages	288
MLD Queries	288
Multicast Client Aging Robustness	289
Multicast Router Discovery	289
MLD Reports	289

MLD Done Messages and Immediate-Leave	290
Topology Change Notification Processing	290
MLD Snooping in Switch Stacks	290
How to Configure IPv6 MLD Snooping	291
Default MLD Snooping Configuration	291
MLD Snooping Configuration Guidelines	291
Enabling or Disabling MLD Snooping on the Switch	292
Enabling or Disabling MLD Snooping on a VLAN	293
Configuring a Static Multicast Group	294
Configuring a Multicast Router Port	295
Enabling MLD Immediate Leave	296
Configuring MLD Snooping Queries	296
Disabling MLD Listener Message Suppression	298
Displaying MLD Snooping Information	299
Configuration Examples for Configuring MLD Snooping	300
Configuring a Static Multicast Group: Example	300
Configuring a Multicast Router Port: Example	300
Enabling MLD Immediate Leave: Example	300
Configuring MLD Snooping Queries: Example	300

CHAPTER 21

Configuring IPv6 Unicast Routing	303
Finding Feature Information	303
Information About Configuring IPv6 Unicast Routing	303
Understanding IPv6	303
IPv6 Addresses	304
Supported IPv6 Unicast Routing Features	304
Unsupported IPv6 Unicast Routing Features	309
IPv6 Feature Limitations	309
Configuring IPv6	309
Default IPv6 Configuration	309
Configuring IPv6 Addressing and Enabling IPv6 Routing	310
Configuring First Hop Security in IPv6	312
Configuring Default Router Preference	323
Configuring IPv6 ICMP Rate Limiting	324

Configuring Cisco Express Forwarding and distributed Cisco Express Forwarding for IPv6	325
Configuring Static Routing for IPv6	325
Configuring RIP for IPv6	328
Configuring OSPF for IPv6	330
Tuning LSA and SPF Timers for OSPFv3 Fast Convergence	332
Configuring LSA and SPF Throttling for OSPFv3 Fast Convergence	333
Configuring EIGRP for IPv6	334
Configuring HSRP for IPv6	334
Enabling HSRP Version 2	335
Enabling an HSRP Group for IPv6	335
Configuring Multi-VRF CE	337
Default Multi-VRF CE Configuration	337
Configuring VRFs	338
Configuring VRF-Aware Services	339
Configuring VRF-Aware Services for Neighbor Discovery	340
Configuring VRF-Aware Services for PING	340
Configuring VRF-Aware Services for HSRP	340
Configuring VRF-Aware Services for Traceroute	341
Configuring VRF-Aware Services for FTP and TFTP	342
Configuring a VPN Routing Session	342
Configuring BGP PE to CE Routing Sessions	344
Multi-VRF CE Configuration Example	345
Displaying Multi-VRF CE Status	349
Displaying IPv6	349
Configuring DHCP for IPv6 Address Assignment	350
Default DHCPv6 Address Assignment Configuration	350
DHCPv6 Address Assignment Configuration Guidelines	350
Enabling DHCPv6 Server Function (CLI)	350
Enabling DHCPv6 Client Function	353
Configuration Examples for IPv6 Unicast Routing	354
Configuring IPv6 Addressing and Enabling IPv6 Routing: Example	354
Configuring Default Router Preference: Example	354
Enabling an HSRP Group for IPv6: Example	354
Enabling DHCPv6 Server Function: Example	355

Enabling DHCPv6 Client Function: Example	355
Configuring IPv6 ICMP Rate Limiting: Example	356
Configuring Static Routing for IPv6: Example	356
Configuring RIP for IPv6: Example	356
Displaying IPv6: Example	356

CHAPTER 22
Implementing IPv6 Multicast 357

Finding Feature Information	357
Information About Implementing IPv6 Multicast Routing	357
IPv6 Multicast Overview	357
IPv6 Multicast Routing Implementation	358
MLD Access Group	358
Explicit Tracking of Receivers	358
IPv6 Multicast User Authentication and Profile Support	358
IPV6 MLD Proxy	359
Protocol Independent Multicast	359
PIM-Sparse Mode	359
IPv6 BSR: Configure RP Mapping	361
PIM-Source Specific Multicast	362
Routable Address Hello Option	364
Bidirectional PIM	364
Static Mroutes	364
MRIB	365
MFIB	365
IPv6 Multicast VRF Lite	365
IPv6 Multicast Process Switching and Fast Switching	365
Multiprotocol BGP for the IPv6 Multicast Address Family	366
NSF and SSO Support In IPv6 Multicast	366
Bandwidth-Based CAC for IPv6 Multicast	367
Implementing IPv6 Multicast	367
Enabling IPv6 Multicast Routing	367
Customizing and Verifying the MLD Protocol	367
Customizing and Verifying MLD on an Interface	367
Implementing MLD Group Limits	369

Configuring Explicit Tracking of Receivers to Track Host Behavior	371
Configuring Multicast User Authentication and Profile Support	371
Enabling MLD Proxy in IPv6	373
Resetting the MLD Traffic Counters	374
Clearing the MLD Interface Counters	375
Configuring PIM	375
Configuring PIM-SM and Displaying PIM-SM Information for a Group Range	375
Configuring PIM Options	377
Configuring Bidirectional PIM and Displaying Bidirectional PIM Information	378
Resetting the PIM Traffic Counters	379
Clearing the PIM Topology Table to Reset the MRIB Connection	380
Configuring a BSR	381
Configuring a BSR and Verifying BSR Information	381
Sending PIM RP Advertisements to the BSR	382
Configuring BSR for Use Within Scoped Zones	383
Configuring BSR Switches to Announce Scope-to-RP Mappings	384
Configuring SSM Mapping	385
Configuring Static Mroutes	386
Using MFIB in IPv6 Multicast	387
Verifying MFIB Operation in IPv6 Multicast	387
Resetting MFIB Traffic Counters	388

PART IV
Layer 2 391

CHAPTER 23
Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling 393

Finding Feature Information	393
Prerequisites for Configuring Tunneling	393
IEEE 802.1Q Tunneling	393
Layer 2 Protocol Tunneling	394
Layer 2 Tunneling for EtherChannels	395
Information about Tunneling	395
IEEE 802.1Q and Layer 2 Protocol Overview	395
IEEE 802.1Q Tunneling	396
IEEE 802.1Q Tunneling Configuration Guidelines	398

Native VLANs	398
System MTU	399
Default IEEE 802.1Q Tunneling Configuration	399
Layer 2 Protocol Tunneling Overview	399
Layer 2 Protocol Tunneling on Ports	401
Default Layer 2 Protocol Tunneling Configuration	402
How to Configure Tunneling	403
Configuring an IEEE 802.1Q Tunneling Port	403
Configuring Layer 2 Protocol Tunneling	405
Configuring the SP Edge Switch	408
Configuring the Customer Device	411
Configuration Examples for IEEE 802.1Q and Layer 2 Protocol Tunneling	414
Example: Configuring an IEEE 802.1Q Tunneling Port	414
Example: Configuring Layer 2 Protocol Tunneling	414
Examples: Configuring the SP Edge and Customer Switches	415
Monitoring Tunneling Status	416
Where to Go Next	416

CHAPTER 24

Configuring Spanning Tree Protocol	417
Finding Feature Information	417
Restrictions for STP	417
Information About Spanning Tree Protocol	418
Spanning Tree Protocol	418
Spanning-Tree Topology and BPDUs	418
Bridge ID, Device Priority, and Extended System ID	420
Port Priority Versus Path Cost	420
Spanning-Tree Interface States	421
How a Device or Port Becomes the Root Device or Root Port	424
Spanning Tree and Redundant Connectivity	424
Spanning-Tree Address Management	425
Accelerated Aging to Retain Connectivity	425
Spanning-Tree Modes and Protocols	425
Supported Spanning-Tree Instances	426
Spanning-Tree Interoperability and Backward Compatibility	426

STP and IEEE 802.1Q Trunks	427
VLAN-Bridge Spanning Tree	427
Default Spanning-Tree Configuration	427
How to Configure Spanning-Tree Features	428
Changing the Spanning-Tree Mode	428
Disabling Spanning Tree	429
Configuring the Root Device	430
Configuring a Secondary Root Device	432
Configuring Port Priority	433
Configuring Path Cost	434
Configuring the Device Priority of a VLAN	435
Configuring the Hello Time	436
Configuring the Forwarding-Delay Time for a VLAN	437
Configuring the Maximum-Aging Time for a VLAN	438
Configuring the Transmit Hold-Count	439
Monitoring Spanning-Tree Status	440

CHAPTER 25**Configuring Multiple Spanning-Tree Protocol 441**

Finding Feature Information	441
Prerequisites for MSTP	441
Restrictions for MSTP	442
Information About MSTP	442
MSTP Configuration	442
MSTP Configuration Guidelines	443
Root Switch	443
Multiple Spanning-Tree Regions	444
IST, CIST, and CST	444
Operations Within an MST Region	445
Operations Between MST Regions	445
IEEE 802.1s Terminology	445
Illustration of MST Regions	446
Hop Count	446
Boundary Ports	447
IEEE 802.1s Implementation	447

Port Role Naming Change	448
Interoperation Between Legacy and Standard Devices	448
Detecting Unidirectional Link Failure	449
Interoperability with IEEE 802.1D STP	449
RSTP Overview	449
Port Roles and the Active Topology	450
Rapid Convergence	450
Synchronization of Port Roles	452
Bridge Protocol Data Unit Format and Processing	453
Topology Changes	454
Protocol Migration Process	455
Default MSTP Configuration	455
About MST-to-PVST+ Interoperability (PVST+ Simulation)	456
About Detecting Unidirectional Link Failure	457
How to Configure MSTP Features	458
Specifying the MST Region Configuration and Enabling MSTP	458
Configuring the Root Device	460
Configuring a Secondary Root Device	461
Configuring Port Priority	462
Configuring Path Cost	464
Configuring the Device Priority	465
Configuring the Hello Time	467
Configuring the Forwarding-Delay Time	468
Configuring the Maximum-Aging Time	469
Configuring the Maximum-Hop Count	470
Specifying the Link Type to Ensure Rapid Transitions	470
Designating the Neighbor Type	472
Restarting the Protocol Migration Process	473
Configuring PVST+ Simulation	474
Enabling PVST+ Simulation on a Port	475
Examples	476
Examples: PVST+ Simulation	476
Examples: Detecting Unidirectional Link Failure	479
Monitoring MST Configuration and Status	480

Feature Information for MSTP 480

CHAPTER 26

Configuring Optional Spanning-Tree Features 481

Finding Feature Information 481

Restriction for Optional Spanning-Tree Features 481

Information About Optional Spanning-Tree Features 481

PortFast 481

BPDU Guard 482

BPDU Filtering 482

UplinkFast 483

Cross-Stack UplinkFast 484

How Cross-Stack UplinkFast Works 485

Events That Cause Fast Convergence 486

BackboneFast 486

EtherChannel Guard 488

Root Guard 489

Loop Guard 490

STP PortFast Port Types 490

Bridge Assurance 491

How to Configure Optional Spanning-Tree Features 493

Enabling PortFast 493

Enabling BPDU Guard 494

Enabling BPDU Filtering 496

Enabling UplinkFast for Use with Redundant Links 497

Disabling UplinkFast 498

Enabling BackboneFast 499

Enabling EtherChannel Guard 500

Enabling Root Guard 501

Enabling Loop Guard 503

Enabling PortFast Port Types 504

Configuring the Default Port State Globally 504

Configuring PortFast Edge on a Specified Interface 505

Configuring a PortFast Network Port on a Specified Interface 506

Enabling Bridge Assurance 507

Examples	508
Examples: Configuring PortFast Edge on a Specified Interface	508
Examples: Configuring a PortFast Network Port on a Specified Interface	509
Example: Configuring Bridge Assurance	510
Monitoring the Spanning-Tree Status	511

CHAPTER 27
Configuring Bidirection Forwarding Detection 513

Finding Feature Information	513
Prerequisites for Bidirectional Forwarding Detection	513
Restrictions for Bidirectional Forwarding Detection	513
Information About Bidirectional Forwarding Detection	514
BFD Operation	514
Neighbor Relationships	514
BFD Detection of Failures	515
BFD Version Interoperability	515
BFD Session Limits	516
BFD Support for Nonbroadcast Media Interfaces	516
BFD Support for Nonstop Forwarding with Stateful Switchover	516
BFD Support for Stateful Switchover	516
BFD Support for Static Routing	517
Benefits of Using BFD for Failure Detection	517
How to Configure Bidirectional Forwarding Detection	518
Configuring BFD Session Parameters on the Interface	518
Configuring BFD Support for Dynamic Routing Protocols	519
Configuring BFD Support for BGP	519
Configuring BFD Support for EIGRP	520
Configuring BFD Support for OSPF	522
Configuring BFD Support for Static Routing	525
Configuring BFD Echo Mode	527
Monitoring and Troubleshooting BFD	529
Configuration Examples for Bidirectional Forwarding Detection	530
Example: Configuring BFD in an EIGRP Network with Echo Mode Enabled by Default	530
Example: Configuring BFD in an OSPF Network	535
Example: Configuring BFD Support for Static Routing	539

CHAPTER 28

Configuring EtherChannels	541
Finding Feature Information	541
Restrictions for EtherChannels	541
Information About EtherChannels	542
EtherChannel Overview	542
EtherChannel Modes	542
EtherChannel on Devices	543
EtherChannel Link Failover	543
Channel Groups and Port-Channel Interfaces	543
Port Aggregation Protocol	544
PAgP Modes	544
PAgP Learn Method and Priority	545
PAgP Interaction with Virtual Switches and Dual-Active Detection	545
PAgP Interaction with Other Features	546
Link Aggregation Control Protocol	546
LACP Modes	546
LACP Interaction with Other Features	547
EtherChannel On Mode	547
Load-Balancing and Forwarding Methods	547
MAC Address Forwarding	547
IP Address Forwarding	548
Load-Balancing Advantages	548
EtherChannel Load Deferral Overview	549
Default EtherChannel Configuration	550
EtherChannel Configuration Guidelines	551
Layer 2 EtherChannel Configuration Guidelines	551
Auto-LAG	552
Auto-LAG Configuration Guidelines	552
How to Configure EtherChannels	553
Configuring Layer 2 EtherChannels	553
Configuring EtherChannel Load-Balancing	555
Configuring Port Channel Load Deferral	556
Configuring the PAgP Learn Method and Priority	558

Configuring LACP Hot-Standby Ports	559
Configuring the LACP System Priority	559
Configuring the LACP Port Priority	560
Configuring the LACP Port Channel Min-Links Feature	561
Configuring LACP Fast Rate Timer	562
Configuring Auto-LAG Globally	564
Configuring Auto-LAG on a Port Interface	564
Configuring Persistence with Auto-LAG	565
Monitoring EtherChannel, PAgP, and LACP Status	566
Configuration Examples for Configuring EtherChannels	567
Configuring Layer 2 EtherChannels: Examples	567
Example: Configuring Port Channel Load Deferral	568
Configuring Auto LAG: Examples	568
Configuring LACP Port Channel Min-Links: Examples	569
Example: Configuring LACP Fast Rate Timer	569

CHAPTER 29

Configuring Link-State Tracking	571
Finding Feature Information	571
Restrictions for Configuring Link-State Tracking	571
Understanding Link-State Tracking	571
How to Configure Link-State Tracking	574
Monitoring Link-State Tracking	575
Configuring Link-State Tracking: Example	575

CHAPTER 30

Configuring Resilient Ethernet Protocol	577
Finding Feature Information	577
Overview of Resilient Ethernet Protocol	577
Link Integrity	579
Fast Convergence	580
VLAN Load Balancing	580
Spanning Tree Interaction	581
REP Ports	582
How to Configure Resilient Ethernet Protocol	582
Default REP Configuration	582

REP Configuration Guidelines	583
Configuring REP Administrative VLAN	584
Configuring a REP Interface	585
Setting Manual Preemption for VLAN Load Balancing	589
Configuring SNMP Traps for REP	590
Monitoring Resilient Ethernet Protocol Configuration	591
Configuration Examples for Resilient Ethernet Protocol	592
Example: Configuring the REP Administrative VLAN	592
Example: Configuring a REP Interface	593
Feature Information for Resilient Ethernet Protocol	594
<hr/>	
CHAPTER 31	Configuring Flex Links and the MAC Address-Table Move Update Feature
	595
Finding Feature Information	595
Restrictions for Configuring Flex Links and MAC Address-Table Move Update	595
Information About Flex Links and MAC Address-Table Move Update	596
Flex Links	596
Flex Links Configuration	596
VLAN Flex Links Load Balancing and Support	597
Multicast Fast Convergence with Flex Links Failover	597
Learning the Other Flex Links Port as the mrouter Port	597
Generating IGMP Reports	597
Leaking IGMP Reports	597
MAC Address-Table Move Update	598
Flex Links VLAN Load Balancing Configuration Guidelines	598
MAC Address-Table Move Update Configuration Guidelines	598
Default Flex Links and MAC Address-Table Move Update Configuration	598
How to Configure Flex Links and the MAC Address-Table Move Update Feature	599
Configuring Flex Links	599
Configuring a Preemption Scheme for a Pair of Flex Links	599
Configuring VLAN Load Balancing on Flex Links	601
Configuring MAC Address-Table Move Update	602
Configuring a Device to Obtain and Process MAC Address-Table Move Update Messages	603
Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update	604
Configuration Examples for Flex Links	604

Configuring Flex Links: Examples 604

Configuring VLAN Load Balancing on Flex Links: Examples 605

Configuring the MAC Address-Table Move Update: Examples 606

Configuring Multicast Fast Convergence with Flex Links Failover: Examples 606

CHAPTER 32 **Configuring UniDirectional Link Detection 609**

Finding Feature Information 609

Restrictions for Configuring UDLD 609

Information About UDLD 610

 Modes of Operation 610

 Normal Mode 610

 Aggressive Mode 610

Methods to Detect Unidirectional Links 611

 Neighbor Database Maintenance 611

 Event-Driven Detection and Echoing 611

 UDLD Reset Options 611

Default UDLD Configuration 612

How to Configure UDLD 612

 Enabling UDLD Globally 612

 Enabling UDLD on an Interface 613

Monitoring and Maintaining UDLD 614

PART V **Stack Manager and High Availability 615**

CHAPTER 33 **Configuring HSRP and VRRP 617**

 Configuring HSRP 617

 Information About Configuring HSRP 617

 HSRP Overview 617

 HSRP Versions 619

 Multiple HSRP 620

 SSO HSRP 620

How to Configure HSRP 621

 Default HSRP Configuration 621

 HSRP Configuration Guidelines 621

Enabling HSRP	621
Configuring HSRP Priority	623
Configuring MHSRP	625
Configuring HSRP Authentication and Timers	631
Enabling HSRP Support for ICMP Redirect Messages	633
Configuring HSRP Groups and Clustering	633
Troubleshooting HSRP	633
Verifying HSRP	634
Verifying HSRP Configurations	634
Configuration Examples for Configuring HSRP	634
Enabling HSRP: Example	634
Configuring HSRP Priority: Example	635
Configuring MHSRP: Example	635
Configuring HSRP Authentication and Timer: Example	635
Configuring HSRP Groups and Clustering: Example	636
Information About VRRP	636
Configuring VRRP	636

CHAPTER 34

Configuring Service Level Agreements	637
Finding Feature Information	637
Restrictions on SLAs	637
Information About SLAs	638
Cisco IOS IP Service Level Agreements (SLAs)	638
Network Performance Measurement with Cisco IOS IP SLAs	639
IP SLA Responder and IP SLA Control Protocol	639
Response Time Computation for IP SLAs	640
IP SLAs Operation Scheduling	641
IP SLA Operation Threshold Monitoring	641
UDP Jitter	642
How to Configure IP SLAs Operations	642
Default Configuration	642
Configuration Guidelines	643
Configuring the IP SLA Responder	643
Implementing IP SLA Network Performance Measurement	645

Analyzing IP Service Levels by Using the UDP Jitter Operation	648
Analyzing IP Service Levels by Using the ICMP Echo Operation	652
Monitoring IP SLA Operations	655
Monitoring IP SLA Operation Examples	656

CHAPTER 35**Configuring Enhanced Object Tracking 659**

Finding Feature Information	659
Information About Enhanced Object Tracking	659
Enhanced Object Tracking Overview	659
Tracking Interface Line-Protocol or IP Routing State	660
Tracked Lists	660
Tracking Other Characteristics	660
IP SLAs Object Tracking	661
Static Route Object Tracking	661
How to Configure Enhanced Object Tracking	661
Configuring Tracking for Line State Protocol or IP Routing State on an Interface	661
Configuring Tracked Lists	662
Configuring a Tracked List with a Weight Threshold	662
Configuring a Tracked List with a Percentage Threshold	664
Configuring HSRP Object Tracking	666
Configuring IP SLAs Object Tracking	668
Configuring Static Route Object Tracking	669
Configuring a Primary Interface for Static Routing	669
Configuring a Primary Interface for DHCP	670
Configuring IP SLAs Monitoring Agent	670
Configuring a Routing Policy and a Default Route	672
Monitoring Enhanced Object Tracking	674

CHAPTER 36**Managing Switch Stacks 675**

Prerequisites for Switch Stacks	675
Restrictions for Switch Stacks	675
Information About Switch Stacks	675
Horizontal Stacking	675
Switch Stack Membership	676

Changes to Switch Stack Membership	677
Stack Member Numbers	678
Stack Member Priority Values	679
Switch Stack Bridge ID and MAC Address	680
Persistent MAC Address on the Switch Stack	680
Active and Standby Switch Election and Reelection	680
Switch Stack Configuration Files	680
Offline Configuration to Provision a Stack Member	681
Effects of Adding a Provisioned Switch to a Switch Stack	682
Effects of Replacing a Provisioned Switch in a Switch Stack	683
Effects of Removing a Provisioned Switch from a Switch Stack	683
Stack Protocol Version	683
Major Stack Protocol Version Number Incompatibility Among Stack-Capable Switches	683
Minor Stack Protocol Version Number Incompatibility Among Stack-Capable Switches	683
Auto-Upgrade	683
Switch Stack Management Connectivity	684
Connectivity to Specific Stack Members	684
Connectivity to the Switch Stack Through an IP Address	684
Connectivity to the Switch Stack Through Console Ports	684
How to Configure a Switch Stack	685
Configuring a Network Port as Stack Port	685
Enabling the Persistent MAC Address Feature	687
Assigning a Stack Member Number	688
Setting the Stack Member Priority Value	690
Provisioning a New Member for a Switch Stack	690
Removing Provisioned Switch Information	691
Troubleshooting the Switch Stack	693
Temporarily Disabling a Stack Port	693
Reenabling a Stack Port While Another Member Starts	694
Monitoring the Device Stack	694
Configuration Examples for Switch Stacks	695
Switch Stack Configuration Scenarios	695
Enabling the Persistent MAC Address Feature: Example	696
Provisioning a New Member for a Switch Stack: Example	697

Configuring a Network Port into a Stack Port: Examples 697
 Additional References for Switch Stacks 698

PART VI

Network Management 701

CHAPTER 37

Configuring Cisco IOS Configuration Engine 703

Prerequisites for Configuring the Configuration Engine 703
 Restrictions for Configuring the Configuration Engine 703
 Information About Configuring the Configuration Engine 703
 Cisco Configuration Engine Software 703
 Configuration Service 704
 Event Service 705
 NameSpace Mapper 705
 Cisco Networking Services IDs and Device Hostnames 705
 ConfigID 706
 DeviceID 706
 Hostname and DeviceID 706
 Hostname, DeviceID, and ConfigID 706
 Cisco IOS CNS Agents 707
 Initial Configuration 707
 Incremental (Partial) Configuration 707
 Synchronized Configuration 708
 Automated CNS Configuration 708
 How to Configure the Configuration Engine 709
 Enabling the CNS Event Agent 709
 Enabling the Cisco IOS CNS Agent 710
 Enabling an Initial Configuration for Cisco IOS CNS Agent 712
 Refreshing DeviceIDs 717
 Enabling a Partial Configuration for Cisco IOS CNS Agent 718
 Monitoring CNS Configurations 720

CHAPTER 38

Configuring the Cisco Discovery Protocol 721

Information About CDP 721
 Cisco Discovery Protocol Overview 721

Default Cisco Discovery Protocol Configuration	722
How to Configure CDP	722
Configuring Cisco Discovery Protocol Characteristics	722
Disabling Cisco Discovery Protocol	724
Enabling Cisco Discovery Protocol	725
Disabling Cisco Discovery Protocol on an Interface	726
Enabling Cisco Discovery Protocol on an Interface	728
Monitoring and Maintaining Cisco Discovery Protocol	729

CHAPTER 39**Configuring Simple Network Management Protocol 731**

Prerequisites for SNMP	731
Restrictions for SNMP	733
Information About SNMP	733
SNMP Overview	733
SNMP Manager Functions	734
SNMP Agent Functions	734
SNMP Community Strings	735
SNMP MIB Variables Access	735
SNMP Notifications	735
SNMP ifIndex MIB Object Values	736
Default SNMP Configuration	736
SNMP Configuration Guidelines	737
How to Configure SNMP	738
Disabling the SNMP Agent	738
Configuring Community Strings	739
Configuring SNMP Groups and Users	741
Configuring SNMP Notifications	744
Setting the Agent Contact and Location Information	749
Limiting TFTP Servers Used Through SNMP	750
Monitoring SNMP Status	752
SNMP Examples	752

CHAPTER 40**Configuring SPAN and RSPAN 755**

Prerequisites for SPAN and RSPAN	755
----------------------------------	-----

Restrictions for SPAN and RSPAN	755
Information About SPAN and RSPAN	757
SPAN and RSPAN	757
Local SPAN	757
Remote SPAN	758
SPAN and RSPAN Concepts and Terminology	759
SPAN and RSPAN Interaction with Other Features	764
Flow-Based SPAN	765
Default SPAN and RSPAN Configuration	766
Configuration Guidelines	766
SPAN Configuration Guidelines	766
RSPAN Configuration Guidelines	766
FSPAN and FRSPAN Configuration Guidelines	767
How to Configure SPAN and RSPAN	767
Creating a Local SPAN Session	767
Creating a Local SPAN Session and Configuring Incoming Traffic	770
Specifying VLANs to Filter	772
Configuring a VLAN as an RSPAN VLAN	774
Creating an RSPAN Source Session	775
Specifying VLANs to Filter	777
Creating an RSPAN Destination Session	779
Creating an RSPAN Destination Session and Configuring Incoming Traffic	781
Configuring an FSPAN Session	783
Configuring an FRSPAN Session	786
Monitoring SPAN and RSPAN Operations	789
SPAN and RSPAN Configuration Examples	789
Example: Configuring Local SPAN	789
Examples: Creating an RSPAN VLAN	790

CHAPTER 41
Configuring RMON 793

Finding Feature Information	793
Information About RMON	793
Understanding RMON	793
How to Configure RMON	794

Default RMON Configuration	794
Configuring RMON Alarms and Events	795
Collecting Group History Statistics on an Interface	797
Collecting Group Ethernet Statistics on an Interface	798
Monitoring RMON Status	800

CHAPTER 42
Configuring Embedded Event Manager 801

Information about Embedded Event Manager	801
Understanding Embedded Event Manager	801
Embedded Event Manager Actions	802
Embedded Event Manager Policies	802
Embedded Event Manager Environment Variables	803
Embedded Event Manager 3.2	803
How to Configure Embedded Event Manager	804
Registering and Defining an Embedded Event Manager Applet	804
Registering and Defining an Embedded Event Manager TCL Script	805
Monitoring Embedded Event Manager	806
Displaying Embedded Event Manager Information	806
Configuration Examples for Embedded Event Manager	807
Example: Generating SNMP Notifications	807
Example: Responding to EEM Events	807
Example: Displaying EEM Environment Variables	807

CHAPTER 43
Configuring Flexible NetFlow 809

Finding Feature Information	809
Prerequisites for NetFlow Lite	809
Restrictions for NetFlow Lite	809
Information About NetFlow Lite	811
NetFlow Lite Overview	811
Flexible NetFlow Components	811
Flow Records	811
Flow Exporters	815
Flow Monitors	816
Flow Samplers	818

- Default Settings **818**
- How to Configure Flexible Netflow **818**
 - Creating a Flow Record **819**
 - Creating a Flow Exporter **821**
 - Creating a Flow Monitor **823**
 - Creating a Sampler **824**
 - Applying a Flow to an Interface **826**
 - Configuring a Bridged NetFlow on a VLAN **827**
 - Configuring Layer 2 NetFlow **828**
- Monitoring Flexible NetFlow **829**
- Configuration Examples for NetFlow Lite **830**
 - Example: Configuring a Flow **830**
- Feature Information for Flexible NetFlow **831**

CHAPTER 44

Configuring Cache Services Using the Web Cache Communication Protocol **833**

- Finding Feature Information **833**
- Prerequisites for WCCP **833**
- Restrictions for WCCP **834**
- Information About WCCP **835**
 - WCCP Overview **835**
 - WCCP Message Exchange **835**
 - WCCP Negotiation **836**
 - MD5 Security **836**
 - Packet Redirection and Service Groups **836**
- How to Configure WCCP **838**
 - Default WCCP Configuration **838**
 - Enabling the Cache Service **838**

PART VII

QoS **845**

CHAPTER 45

Configuring QoS **847**

- Finding Feature Information **847**
- Prerequisites for QoS **847**
 - QoS ACL Guidelines **847**

Policing Guidelines	848
General QoS Guidelines	848
Restrictions for QoS	849
Information About QoS	850
QoS Implementation	850
Layer 2 Frame Prioritization Bits	851
Layer 3 Packet Prioritization Bits	851
End-to-End QoS Solution Using Classification	851
QoS Basic Model	851
Actions at Ingress Port	852
Actions at Egress Port	852
Classification Overview	853
Policing and Marking Overview	857
Mapping Tables Overview	858
Queueing and Scheduling Overview	859
Queueing and Scheduling on Egress Queues	861
Packet Modification	865
Standard QoS Default Configuration	865
Default Egress Queue Configuration	866
Default Mapping Table Configuration	869
DSCP Maps	869
Default CoS-to-DSCP Map	869
Default IP-Precedence-to-DSCP Map	869
Default DSCP-to-CoS Map	870
How to Configure QoS	870
Enabling QoS Globally	870
Enabling VLAN-Based QoS on Physical Ports	871
Configuring a QoS Policy	872
Classifying Traffic by Using ACLs	873
Classifying Traffic by Using Class Maps	880
Classifying Traffic by Using Class Maps and Filtering IPv6 Traffic	883
Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps	884
Classifying, Policing, and Marking Traffic by Using Aggregate Policers	889
Configuring DSCP Maps	891

Configuring the CoS-to-DSCP Map	891
Configuring the IP-Precedence-to-DSCP Map	892
Configuring the Policed-DSCP Map	894
Configuring the DSCP-to-CoS Map	895
Configuring the DSCP-to-DSCP-Mutation Map	896
Configuring Egress Queue Characteristics	898
Configuration Guidelines	898
Allocating Buffer Space to and Setting WTD Thresholds for an Egress Queue-Set	898
Mapping DSCP or CoS Values to an Egress Queue and to a Threshold ID	901
Configuring SRR Shaped Weights on Egress Queues	904
Configuring SRR Shared Weights on Egress Queues	906
Configuring the Egress Expedite Queue	907
Limiting the Bandwidth on an Egress Interface	909
Monitoring Standard QoS	910
Configuration Examples for QoS	911
Example: Configuring Port to the DSCP-Trusted State and Modifying the DSCP-to-DSCP-Mutation Map	911
Examples: Classifying Traffic by Using ACLs	911
Examples: Classifying Traffic by Using Class Maps	912
Examples: Classifying, Policing, and Marking Traffic on Physical Ports Using Policy Maps	913
Examples: Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps	915
Examples: Classifying, Policing, and Marking Traffic by Using Aggregate Policers	917
Examples: Configuring DSCP Maps	917
Examples: Configuring Egress Queue Characteristics	919
Where to Go Next	920
<hr/>	
CHAPTER 46	Configuring Auto-QoS 921
Finding Feature Information	921
Prerequisites for Auto-QoS	921
Information about Configuring Auto-QoS	922
Auto-QoS Overview	922
Generated Auto-QoS Configuration	922
VoIP Device Specifics	923
Enhanced Auto-QoS for Video, Trust, and Classification	924

Auto-QoS Configuration Migration	924
Auto-QoS Configuration Guidelines	924
Auto-QoS VoIP Considerations	925
Auto-QoS Enhanced Considerations	925
Effects of Auto-QoS on Running Configuration	925
How to Configure Auto-QoS	926
Configuring Auto-QoS	926
Enabling Auto-QoS	926
Troubleshooting Auto-QoS	928
Monitoring Auto-QoS	928
Configuration Examples for Auto-QoS	929
Examples: Global Auto-QoS Configuration	929
Examples: Auto-QoS Generated Configuration for VoIP Devices	931
Examples: Auto-QoS Generated Configuration for VoIP Devices	933
Examples: Auto-QoS Generated Configuration For Enhanced Video, Trust, and Classify Devices	934
Where to Go Next for Auto-QoS	937

PART VIII
Routing 939

CHAPTER 47
Configuring IP Unicast Routing 941

Finding Feature Information	941
Information About Configuring IP Unicast Routing	941
Information About IP Routing	942
Types of Routing	942
How to Configure IP Routing	942
How to Configure IP Addressing	943
Default IP Addressing Configuration	944
Assigning IP Addresses to Network Interfaces	945
Using Subnet Zero	946
Classless Routing	947
Disabling Classless Routing	949
Configuring Address Resolution Methods	950
Address Resolution	950
Defining a Static ARP Cache	951

Setting ARP Encapsulation	952
Enabling Proxy ARP	953
Routing Assistance When IP Routing is Disabled	954
Proxy ARP	955
Proxy ARP	955
Default Gateway	955
ICMP Router Discovery Protocol	956
ICMP Router Discovery Protocol (IRDP)	956
Configuring Broadcast Packet Handling	958
Broadcast Packet Handling	958
Enabling Directed Broadcast-to-Physical Broadcast Translation	959
UDP Broadcast Packets and Protocols	960
Forwarding UDP Broadcast Packets and Protocols	960
Establishing an IP Broadcast Address	962
IP Broadcast Flooding	963
Flooding IP Broadcasts	964
Monitoring and Maintaining IP Addressing	965
How to Configure IP Unicast Routing	966
Enabling IP Unicast Routing	966
Example of Enabling IP Unicast Routing	967
Information About RIP	967
How to Configure RIP	968
Default RIP Configuration	968
Configuring Basic RIP Parameters	968
Configuring RIP Authentication	970
Summary Addresses and Split Horizon	972
Configuring Summary Addresses and Split Horizon	972
Configuring Split Horizon	973
Configuration Example for Summary Addresses and Split Horizon	975
Information About OSPF	975
How to Configure OSPF	976
Default OSPF Configuration	976
Configuring Basic OSPF Parameters	978
Example: Configuring Basic OSPF Parameters	979

Configuring OSPF Interfaces	979
OSPF Area Parameters	981
Configuring OSPF Area Parameters	982
Other OSPF Parameters	983
Configuring Other OSPF Parameters	984
LSA Group Pacing	986
Changing LSA Group Pacing	987
Loopback Interfaces	987
Configuring a Loopback Interface	988
Monitoring OSPF	988
Information About EIGRP	989
EIGRP Features	989
EIGRP Components	990
How to Configure EIGRP	991
Default EIGRP Configuration	991
EIGRP Nonstop Forwarding	992
Configuring Basic EIGRP Parameters	993
Configuring EIGRP Interfaces	995
Configuring EIGRP Route Authentication	996
EIGRP Stub Routing	998
Monitoring and Maintaining EIGRP	999
Information About Multi-VRF CE	999
Understanding Multi-VRF CE	1000
Network Topology	1000
Packet-Forwarding Process	1001
Network Components	1001
VRF-Aware Services	1002
How to Configure Multi-VRF CE	1002
Default Multi-VRF CE Configuration	1002
Multi-VRF CE Configuration Guidelines	1003
Configuring VRFs	1003
Configuring VRF-Aware Services	1005
Configuring VRF-Aware Services for ARP	1005
Configuring VRF-Aware Services for Ping	1006

Configuring VRF-Aware Services for SNMP	1006
Configuring VRF-Aware Services for HSRP	1007
Configuring VRF-Aware Services for uRPF	1008
Configuring VRF-Aware RADIUS	1009
Configuring VRF-Aware Services for Syslog	1009
Configuring VRF-Aware Services for Traceroute	1010
Configuring VRF-Aware Services for FTP and TFTP	1010
Configuring Multicast VRFs	1011
Configuring a VPN Routing Session	1013
Configuring BGP PE to CE Routing Sessions	1014
Multi-VRF CE Configuration Example	1016
Monitoring Multi-VRF CE	1019
Configuring Unicast Reverse Path Forwarding	1019
Protocol-Independent Features	1020
Distributed Cisco Express Forwarding	1020
Information About Cisco Express Forwarding	1020
How to Configure Cisco Express Forwarding	1020
Number of Equal-Cost Routing Paths	1022
Information About Equal-Cost Routing Paths	1022
How to Configure Equal-Cost Routing Paths	1022
Static Unicast Routes	1023
Information About Static Unicast Routes	1023
Configuring Static Unicast Routes	1024
Default Routes and Networks	1025
Information About Default Routes and Networks	1025
How to Configure Default Routes and Networks	1026
Route Maps to Redistribute Routing Information	1026
Information About Route Maps	1026
How to Configure a Route Map	1027
How to Control Route Distribution	1030
Policy-Based Routing	1032
Information About Policy-Based Routing	1032
How to Configure PBR	1033
Filtering Routing Information	1036

Setting Passive Interfaces	1036
Controlling Advertising and Processing in Routing Updates	1037
Filtering Sources of Routing Information	1038
Managing Authentication Keys	1039
Prerequisites	1039
How to Configure Authentication Keys	1039
Monitoring and Maintaining the IP Network	1041

CHAPTER 48 **Configuring Policy-Based Routing (PBR)** **1043**

Policy-Based Routing	1043
Information About Policy-Based Routing	1043
How to Configure PBR	1044
Feature Information for Configuring PBR	1047

CHAPTER 49 **Configuring EIGRP Stub Routing** **1049**

EIGRP Stub Routing	1049
Information About EIGRP Stub Routing	1049
EIGRP Stub Routing	1049
How to Configure EIGRP Stub Routing	1052
Configuring the EIGRP Stub Routing Autonomous System Configuration	1052
Configuring the EIGRP Stub Routing Named Configuration	1053
Configuration Examples for EIGRP Stub Routing	1054
Example: EIGRP Stub Routing—Autonomous System Configuration	1054
Example: EIGRP Stub Routing—Named Configuration	1056
Additional References	1057
Feature Information for EIGRP Stub Routing	1058

PART IX **Security** **1059**

CHAPTER 50 **Security Features Overview** **1061**

Security Features Overview	1061
----------------------------	------

CHAPTER 51 **Preventing Unauthorized Access** **1065**

Preventing Unauthorized Access	1065
--------------------------------	------

CHAPTER 52	Controlling Switch Access with Passwords and Privilege Levels	1067
	Restrictions for Controlling Switch Access with Passwords and Privileges	1067
	Restrictions and Guidelines for Reversible Password Types	1067
	Restrictions and Guidelines for Irreversible Password Types	1067
	Information About Passwords and Privilege Levels	1068
	Default Password and Privilege Level Configuration	1068
	Additional Password Security	1068
	Password Recovery	1069
	Terminal Line Telnet Configuration	1069
	Username and Password Pairs	1069
	Privilege Levels	1069
	How to Control Switch Access with Passwords and Privilege Levels	1070
	Setting or Changing a Static Enable Password	1070
	Protecting Enable and Enable Secret Passwords with Encryption	1071
	Configuring Masked Secret Password	1073
	Disabling Password Recovery	1074
	Setting a Telnet Password for a Terminal Line	1075
	Configuring Username and Password Pairs	1077
	Setting the Privilege Level for a Command	1078
	Changing the Default Privilege Level for Lines	1080
	Logging into and Exiting a Privilege Level	1081
	Monitoring Switch Access	1082
	Configuration Examples for Setting Passwords and Privilege Levels	1082
	Example: Setting or Changing a Static Enable Password	1082
	Example: Protecting Enable and Enable Secret Passwords with Encryption	1082
	Example: Configuring Masked Secret Password	1082
	Example: Setting a Telnet Password for a Terminal Line	1082
	Example: Setting the Privilege Level for a Command	1083

CHAPTER 53	Configuring TACACS+	1085
	Finding Feature Information	1085
	Prerequisites for TACACS+	1085
	Information About TACACS+	1086

TACACS+ and Switch Access	1086
TACACS+ Overview	1087
TACACS+ Operation	1088
Method List	1088
TACACS+ Configuration Options	1089
TACACS+ Login Authentication	1089
TACACS+ Authorization for Privileged EXEC Access and Network Services	1089
TACACS+ Accounting	1090
Default TACACS+ Configuration	1090
How to Configure Switch Access with TACACS+	1090
Identifying the TACACS+ Server Host and Setting the Authentication Key	1090
Configuring TACACS+ Login Authentication	1092
Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services	1094
Starting TACACS+ Accounting	1096
Establishing a Session with a Router if the AAA Server is Unreachable	1097
Monitoring TACACS+	1097

CHAPTER 54**Configuring RADIUS 1099**

Finding Feature Information	1099
Prerequisites for Configuring RADIUS	1099
Restrictions for Configuring RADIUS	1100
Information about RADIUS	1100
RADIUS and Switch Access	1100
RADIUS Overview	1101
RADIUS Operation	1101
RADIUS Change of Authorization	1102
Change-of-Authorization Requests	1104
CoA Request Response Code	1105
CoA Request Commands	1106
Default RADIUS Configuration	1108
RADIUS Server Host	1108
RADIUS Login Authentication	1109
AAA Server Groups	1109
AAA Authorization	1110

RADIUS Accounting	1110
Vendor-Specific RADIUS Attributes	1110
Vendor-Proprietary RADIUS Server Communication	1121
How to Configure RADIUS	1121
Identifying the RADIUS Server Host	1121
Configuring RADIUS Login Authentication	1123
Defining AAA Server Groups	1126
Configuring RADIUS Authorization for User Privileged Access and Network Services	1128
Starting RADIUS Accounting	1129
Configuring Settings for All RADIUS Servers	1130
Configuring the Device to Use Vendor-Specific RADIUS Attributes	1132
Configuring the Device for Vendor-Proprietary RADIUS Server Communication	1133
Configuring CoA on the Device	1135
Monitoring CoA Functionality	1137
Configuration Examples for Controlling Switch Access with RADIUS	1138
Examples: Identifying the RADIUS Server Host	1138
Example: Using Two Different RADIUS Group Servers	1138
Examples: Configuring the Switch to Use Vendor-Specific RADIUS Attributes	1138
Example: Configuring the Switch for Vendor-Proprietary RADIUS Server Communication	1139

CHAPTER 55
Configuring Kerberos 1141

Prerequisites for Controlling Switch Access with Kerberos	1141
Information about Kerberos	1141
Kerberos and Switch Access	1141
Kerberos Overview	1142
Kerberos Operation	1144
Authenticating to a Boundary Switch	1144
Obtaining a TGT from a KDC	1144
Authenticating to Network Services	1145
How to Configure Kerberos	1145
Monitoring the Kerberos Configuration	1145

CHAPTER 56
Configuring Local Authentication and Authorization 1147

How to Configure Local Authentication and Authorization	1147
---------------------------------------------------------	------

Configuring the Switch for Local Authentication and Authorization	1147
Monitoring Local Authentication and Authorization	1149

CHAPTER 57**Configuring Secure Shell 1151**

Finding Feature Information	1151
Prerequisites for Configuring Secure Shell	1151
Restrictions for Configuring Secure Shell	1152
Information About Configuring Secure Shell	1152
SSH and Device Access	1152
SSH Servers, Integrated Clients, and Supported Versions	1153
SSH Configuration Guidelines	1153
Secure Copy Protocol Overview	1154
Secure Copy Protocol	1154
How to Configure SSH	1154
Setting Up the Switch to Run SSH	1154
Configuring the SSH Server	1156
Monitoring the SSH Configuration and Status	1158

CHAPTER 58**Configuring SSH File Transfer Protocol 1159**

Prerequisites for SSH File Transfer Protocol	1159
Restrictions for SSH File Transfer Protocol	1159
Information About SSH File Transfer Protocol	1159
How to Configure SSH File Transfer Protocol	1160
Configuring SFTP	1160
Perform an SFTP Copy Operation	1161
Example: Configuring SSH File Transfer Protocol	1161
Additional References	1162
Feature Information for SSH File Transfer Protocol	1162

CHAPTER 59**X.509v3 Certificates for SSH Authentication 1163**

Prerequisites for X.509v3 Certificates for SSH Authentication	1163
Restrictions for X.509v3 Certificates for SSH Authentication	1163
Information About X.509v3 Certificates for SSH Authentication	1164
X.509v3 Certificates for SSH Authentication Overview	1164

Server and User Authentication Using X.509v3	1164
OCSP Response Stapling	1164
How to Configure X.509v3 Certificates for SSH Authentication	1165
Configuring Digital Certificates for Server Authentication	1165
Configuring Digital Certificates for User Authentication	1166
Verifying the Server and User Authentication Using Digital Certificates	1168
Configuration Examples for X.509v3 Certificates for SSH Authentication	1172
Example: Configuring Digital Certificates for Server Authentication	1172
Example: Configuring Digital Certificate for User Authentication	1173
Additional References for X.509v3 Certificates for SSH Authentication	1173
Feature Information for X.509v3 Certificates for SSH Authentication	1174

CHAPTER 60**Configuring Secure Socket Layer HTTP 1175**

Finding Feature Information	1175
Information about Secure Sockets Layer (SSL) HTTP	1175
Secure HTTP Servers and Clients Overview	1175
Certificate Authority Trustpoints	1176
CipherSuites	1177
Default SSL Configuration	1178
SSL Configuration Guidelines	1178
How to Configure Secure HTTP Servers and Clients	1178
Configuring a CA Trustpoint	1178
Configuring the Secure HTTP Server	1181
Configuring the Secure HTTP Client	1184
Monitoring Secure HTTP Server and Client Status	1185

CHAPTER 61**Certification Authority Interoperability 1187**

Prerequisites For Certification Authority	1187
Restrictions for Certification Authority	1187
Information About Certification Authority	1187
CA Supported Standards	1187
Purpose of CAs	1188
Implementing IPsec Without CAs	1189
Implementing IPsec With CAs	1189

Implementing IPsec with Multiple Root CAs	1189
How CA Certificates Are Used by IPsec Devices	1190
Registration Authorities	1190
How to Configure Certification Authority	1190
Managing NVRAM Memory Usage	1190
Configuring the Device Host Name and IP Domain Name	1192
Generating an RSA Key Pair	1192
Declaring a Certification Authority	1193
Configuring a Root CA (Trusted Root)	1195
Authenticating the CA	1196
Requesting Signed Certificates	1197
Monitoring and Maintaining Certification Authority	1198
Requesting a Certificate Revocation List	1198
Querying a Certification Revocation List	1199
Deleting RSA Keys from a Device	1199
Deleting Public Keys for a Peer	1200
Deleting Certificates from the Configuration	1201
Viewing Keys and Certificates	1202

CHAPTER 62

Access Control List Overview	1205
Information About Access Control Lists	1205
Definition of an Access List	1205
Functions of an Access Control List	1206
Purpose of IP Access Lists	1206
Reasons to Configure ACLs	1206
Software Processing of an Access List	1207
Access List Rules	1207
Helpful Hints for Creating IP Access Lists	1208
IP Packet Fields You Can Filter to Control Access	1209
Source and Destination Addresses	1209
Wildcard Mask for Addresses in an Access List	1209
Access List Sequence Numbers	1210
ACL Supported Types	1210
Supported ACLs	1211

ACL Precedence	1211
Port ACLs	1211
Router ACLs	1212
Access Control Entries	1213
ACEs and Fragmented and Unfragmented Traffic	1213
ACEs and Fragmented and Unfragmented Traffic Examples	1213

CHAPTER 63**IPv4 ACLs 1215**

Finding Feature Information	1215
Restrictions for Configuring IPv4 Access Control Lists	1215
Information about Network Security with ACLs	1216
Cisco TrustSec and ACLs	1216
ACL Overview	1217
Access Control Entries	1217
ACL Supported Types	1217
Supported ACLs	1217
ACL Precedence	1217
Port ACLs	1218
Router ACLs	1219
VLAN Maps	1220
ACEs and Fragmented and Unfragmented Traffic	1220
ACEs and Fragmented and Unfragmented Traffic Examples	1221
Standard and Extended IPv4 ACLs	1221
IPv4 ACL Switch Unsupported Features	1222
Access List Numbers	1222
Numbered Standard IPv4 ACLs	1223
Numbered Extended IPv4 ACLs	1223
Named IPv4 ACLs	1224
ACL Logging	1224
Smart Logging	1225
Hardware and Software Treatment of IP ACLs	1225
VLAN Map Configuration Guidelines	1226
VLAN Maps with Router ACLs	1226
VLAN Maps and Router ACL Configuration Guidelines	1226

VACL Logging	1227
Time Ranges for ACLs	1227
IPv4 ACL Interface Considerations	1228
How to Configure ACLs	1228
Configuring IPv4 ACLs	1228
Creating a Numbered Standard ACL	1229
Creating a Numbered Extended ACL (CLI)	1230
Creating Named Standard ACLs	1234
Creating Extended Named ACLs	1235
Configuring Time Ranges for ACLs	1237
Applying an IPv4 ACL to a Terminal Line	1238
Applying an IPv4 ACL to an Interface (CLI)	1240
Creating Named MAC Extended ACLs	1241
Applying a MAC ACL to a Layer 2 Interface	1243
Configuring VLAN Maps	1244
Creating a VLAN Map	1246
Applying a VLAN Map to a VLAN	1248
Configuring VACL Logging	1249
Monitoring IPv4 ACLs	1250
Configuration Examples for ACLs	1251
Examples: Using Time Ranges with ACLs	1251
Examples: Including Comments in ACLs	1251
Examples: Troubleshooting ACLs	1252
IPv4 ACL Configuration Examples	1253
ACLs in a Small Networked Office	1254
Examples: ACLs in a Small Networked Office	1254
Example: Numbered ACLs	1255
Examples: Extended ACLs	1255
Examples: Named ACLs	1256
Examples: Time Range Applied to an IP ACL	1257
Examples: Configuring Commented IP ACL Entries	1257
Examples: ACL Logging	1258
Configuration Examples for ACLs and VLAN Maps	1259
Example: Creating an ACL and a VLAN Map to Deny a Packet	1259

Example: Creating an ACL and a VLAN Map to Permit a Packet	1259
Example: Default Action of Dropping IP Packets and Forwarding MAC Packets	1259
Example: Default Action of Dropping MAC Packets and Forwarding IP Packets	1260
Example: Default Action of Dropping All Packets	1260
Configuration Examples for Using VLAN Maps in Your Network	1261
Example: Wiring Closet Configuration	1261
Example: Restricting Access to a Server on Another VLAN	1262
Example: Denying Access to a Server on Another VLAN	1262
Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs	1263
Example: ACLs and Switched Packets	1263
Example: ACLs and Bridged Packets	1263
Example: ACLs and Routed Packets	1264
Example: ACLs and Multicast Packets	1265
Feature Information for IPv4 Access Control Lists	1265

CHAPTER 64

IPv6 ACLs 1267

Finding Feature Information	1267
IPv6 ACLs Overview	1267
Interactions with Other Features and Switches	1268
Restrictions for IPv6 ACLs	1268
Default Configuration for IPv6 ACLs	1269
Configuring IPv6 ACLs	1269
Attaching an IPv6 ACL to an Interface	1273
Monitoring IPv6 ACLs	1274

CHAPTER 65

Configuring DHCP 1277

Restrictions for DHCP	1277
Information About DHCP	1277
DHCP Server	1277
DHCP Relay Agent	1277
DHCP Snooping	1278
Option-82 Data Insertion	1279
Cisco IOS DHCP Server Database	1282
DHCP Snooping Binding Database	1282

How to Configure DHCP Features	1284
Default DHCP Snooping Configuration	1284
DHCP Snooping Configuration Guidelines	1285
Configuring the DHCP Server	1285
Configuring the DHCP Relay Agent	1285
Specifying the Packet Forwarding Address	1286
Prerequisites for Configuring DHCP Snooping and Option 82	1288
Enabling DHCP Snooping and Option 82	1289
Enabling the Cisco IOS DHCP Server Database	1292
Monitoring DHCP Snooping Information	1292
Configuring DHCP Server Port-Based Address Allocation	1293
Information About Configuring DHCP Server Port-Based Address Allocation	1293
Default Port-Based Address Allocation Configuration	1293
Port-Based Address Allocation Configuration Guidelines	1293
Enabling the DHCP Snooping Binding Database Agent	1294
Enabling DHCP Server Port-Based Address Allocation	1296
Monitoring DHCP Server Port-Based Address Allocation	1297

CHAPTER 66

Configuring IP Source Guard	1299
Information About IP Source Guard	1299
IP Source Guard	1299
IP Source Guard for Static Hosts	1299
IP Source Guard Configuration Guidelines	1300
How to Configure IP Source Guard	1301
Enabling IP Source Guard	1301
Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port	1303
Monitoring IP Source Guard	1304

CHAPTER 67

Configuring Dynamic ARP Inspection	1307
Restrictions for Dynamic ARP Inspection	1307
Understanding Dynamic ARP Inspection	1308
Interface Trust States and Network Security	1310
Rate Limiting of ARP Packets	1311
Relative Priority of ARP ACLs and DHCP Snooping Entries	1311

Logging of Dropped Packets	1311
Default Dynamic ARP Inspection Configuration	1312
Relative Priority of ARP ACLs and DHCP Snooping Entries	1312
Configuring ARP ACLs for Non-DHCP Environments	1312
Configuring Dynamic ARP Inspection in DHCP Environments	1315
Limiting the Rate of Incoming ARP Packets	1317
Performing Dynamic ARP Inspection Validation Checks	1319
Monitoring DAI	1321
Verifying the DAI Configuration	1321

CHAPTER 68

Configuring IEEE 802.1x Port-Based Authentication	1323
Information About 802.1x Port-Based Authentication	1323
Port-Based Authentication Process	1324
Port-Based Authentication Initiation and Message Exchange	1326
Authentication Manager for Port-Based Authentication	1327
Port-Based Authentication Methods	1327
Per-User ACLs and Filter-Ids	1328
Port-Based Authentication Manager CLI Commands	1329
Ports in Authorized and Unauthorized States	1330
802.1x Host Mode	1331
802.1x Multiple Authentication Mode	1332
Multi-auth Per User VLAN assignment	1332
MAC Move	1333
MAC Replace	1334
802.1x Accounting	1334
802.1x Accounting Attribute-Value Pairs	1335
802.1x Readiness Check	1336
Switch-to-RADIUS-Server Communication	1336
802.1x Authentication with VLAN Assignment	1336
802.1x Authentication with Per-User ACLs	1338
802.1x Authentication with Downloadable ACLs and Redirect URLs	1339
Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL	1340
Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs	1341
VLAN ID-Based MAC Authentication	1341

802.1x Authentication with Guest VLAN	1342
802.1x Authentication with Restricted VLAN	1343
802.1x Authentication with Inaccessible Authentication Bypass	1343
Inaccessible Authentication Bypass Support on Multiple-Authentication Ports	1344
Inaccessible Authentication Bypass Authentication Results	1344
Inaccessible Authentication Bypass Feature Interactions	1344
802.1x Critical Voice VLAN	1345
802.1x User Distribution	1346
802.1x User Distribution Configuration Guidelines	1346
IEEE 802.1x Authentication with Voice VLAN Ports	1346
IEEE 802.1x Authentication with Port Security	1347
IEEE 802.1x Authentication with Wake-on-LAN	1347
IEEE 802.1x Authentication with MAC Authentication Bypass	1348
Network Admission Control Layer 2 IEEE 802.1x Validation	1349
Flexible Authentication Ordering	1349
Open1x Authentication	1350
Multidomain Authentication	1350
Limiting Login for Users	1351
802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)	1352
Voice Aware 802.1x Security	1353
Common Session ID	1354
How to Configure 802.1x Port-Based Authentication	1354
Default 802.1x Authentication Configuration	1354
802.1x Authentication Configuration Guidelines	1356
802.1x Authentication	1356
VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass	1356
MAC Authentication Bypass	1357
Maximum Number of Allowed Devices Per Port	1358
Configuring 802.1x Readiness Check	1358
Configuring Voice Aware 802.1x Security	1360
Configuring 802.1x Violation Modes	1361
Configuring 802.1x Authentication	1363
Configuring 802.1x Port-Based Authentication	1364
Configuring the Switch-to-RADIUS-Server Communication	1366

Configuring the Host Mode	1368
Configuring Periodic Re-Authentication	1369
Changing the Quiet Period	1370
Changing the Switch-to-Client Retransmission Time	1371
Setting the Switch-to-Client Frame-Retransmission Number	1373
Setting the Re-Authentication Number	1374
Enabling MAC Move	1375
Disabling MAC Move	1376
Enabling MAC Replace	1377
Configuring 802.1x Accounting	1378
Configuring a Guest VLAN	1380
Configuring a Restricted VLAN	1381
Configuring Number of Authentication Attempts on a Restricted VLAN	1383
Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN	1384
Example of Configuring Inaccessible Authentication Bypass	1388
Configuring 802.1x Authentication with WoL	1388
Configuring MAC Authentication Bypass	1389
Formatting a MAC Authentication Bypass Username and Password	1390
Configuring 802.1x User Distribution	1391
Example of Configuring VLAN Groups	1392
Configuring NAC Layer 2 802.1x Validation	1393
Configuring Limiting Login for Users	1395
Configuring an Authenticator Switch with NEAT	1396
Configuring a Supplicant Switch with NEAT	1398
Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs	1401
Configuring Downloadable ACLs	1401
Configuring a Downloadable Policy	1402
Configuring VLAN ID-based MAC Authentication	1405
Configuring Flexible Authentication Ordering	1406
Configuring Open 1x	1407
Disabling 802.1x Authentication on the Port	1409
Resetting the 802.1x Authentication Configuration to the Default Values	1410
Monitoring 802.1x Statistics and Status	1411

CHAPTER 69

Configuring MACsec Encryption	1413
Finding Feature Information	1413
Information About MACsec Encryption	1413
Media Access Control Security and MACsec Key Agreement	1414
MKA Policies	1415
Virtual Ports	1415
MACsec and Stacking	1415
MACsec, MKA and 802.1x Host Modes	1416
Information About MACsec MKA using EAP-TLS	1417
Prerequisites for MACsec MKA using EAP-TLS	1417
Limitations for MACsec MKA using EAP-TLS	1418
Cisco TrustSec Overview	1418
Configuring MKA and MACsec	1420
Default MACsec MKA Configuration	1420
Configuring an MKA Policy	1420
Configuring MACsec on an Interface	1421
Configuring MACsec MKA using PSK	1423
Configuring MACsec MKA on an Interface using PSK	1424
Configuring MACsec MKA using EAP-TLS	1425
Remote Authentication	1426
Generating Key Pairs	1426
Configuring Enrollment using SCEP	1426
Configuring Enrollment Manually	1428
Enabling 802.1x Authentication and Configuring AAA	1429
Configuring EAP-TLS Profile and 802.1x Credentials	1430
Applying the 802.1x MACsec MKA Configuration on Interfaces	1431
Local Authentication	1432
Configuring the EAP Credentials using Local Authentication	1432
Configuring the Local EAP-TLS Authentication and Authorization Profile	1432
Configuring Enrollment using SCEP	1433
Configuring Enrollment Manually	1434
Configuring EAP-TLS Profile and 802.1x Credentials	1436
Applying the 802.1x MKA MACsec Configuration on Interfaces	1436

- Verifying MACsec MKA using EAP-TLS 1437
- Configuring Cisco TrustSec MACsec 1439
 - Configuring Cisco TrustSec Credentials on the Switch 1439
 - Configuring Cisco TrustSec Switch-to-Switch Link Security in 802.1x Mode 1440
 - Configuring Cisco TrustSec Switch-to-Switch Link Security in Manual Mode 1442
- Configuration Examples for Configuring MACsec Encryption 1444
 - Example: Configuring MACsec on an Interface 1444
 - Configuration Examples for MACsec MKA using EAP-TLS 1447
 - Example: Enrolling the Certificate 1447
 - Example: Enabling 802.1x Authentication and AAA Configuration 1448
 - Example: Configuring EAP-TLS Profile and 802.1X Credentials 1448
 - Example: Applying 802.1X, PKI, and MACsec Configuration on the Interface 1448
 - Cisco TrustSec Switch-to-Switch Link Security Configuration Example 1449

CHAPTER 70

Web-Based Authentication 1451

- Finding Feature Information 1451
- Web-Based Authentication Overview 1451
 - Device Roles 1452
 - Host Detection 1453
 - Session Creation 1453
 - Authentication Process 1454
 - Local Web Authentication Banner 1454
 - Web Authentication Customizable Web Pages 1457
 - Guidelines 1457
 - Authentication Proxy Web Page Guidelines 1458
 - Redirection URL for Successful Login Guidelines 1459
 - Web-based Authentication Interactions with Other Features 1459
 - Port Security 1459
 - LAN Port IP 1459
 - Gateway IP 1459
 - ACLs 1459
 - Context-Based Access Control 1460
 - EtherChannel 1460
- How to Configure Web-Based Authentication 1460

Default Web-Based Authentication Configuration	1460
Web-Based Authentication Configuration Guidelines and Restrictions	1460
Configuring the Authentication Rule and Interfaces	1462
Configuring AAA Authentication	1463
Configuring Switch-to-RADIUS-Server Communication	1464
Configuring the HTTP Server	1465
Customizing the Authentication Proxy Web Pages	1467
Specifying a Redirection URL for Successful Login	1468
Configuring Web-Based Authentication Parameters	1469
Configuring a Web-Based Authentication Local Banner	1470
Configuring Web-Based Authentication without SVI	1471
Configuring Web-Based Authentication with VRF Aware	1472
Removing Web-Based Authentication Cache Entries	1474
Verifying Web-Based Authentication Status	1474

CHAPTER 71**Auto Identity 1477**

Information About Auto Identity	1477
Auto Identity Overview	1477
Auto Identity Global Template	1478
Auto Identity Interface Templates	1478
Auto Identity Built-in Policies	1479
Auto Identity Class Maps Templates	1480
Auto Identity Parameter Maps	1480
Auto Identity Service Templates	1480
How to Configure Auto Identity	1481
Configuring Auto Identity Globally	1481
Configuring Auto Identity at an Interface Level	1482
Configuration Examples for Auto Identity	1483
Example: Configuring Auto Identity Globally	1483
Example: Configuring Auto Identity at an Interface Level	1484
Verifying Auto Identity	1484
Feature Information for Auto Identity	1487

CHAPTER 72**Configuring Port-Based Traffic Control 1489**

Finding Feature Information	1489
Information About Storm Control	1490
Storm Control	1490
How Traffic Activity is Measured	1490
Traffic Patterns	1491
How to Configure Storm Control	1491
Configuring Storm Control and Threshold Levels	1491
Configuring Small-Frame Arrival Rate	1494
Information About Protected Ports	1496
Protected Ports	1496
Default Protected Port Configuration	1497
Protected Ports Guidelines	1497
How to Configure Protected Ports	1497
Configuring a Protected Port	1497
Monitoring Protected Ports	1498
Where to Go Next	1499
Information About Port Blocking	1499
Port Blocking	1499
How to Configure Port Blocking	1499
Blocking Flooded Traffic on an Interface	1499
Monitoring Port Blocking	1501
Prerequisites for Port Security	1501
Restrictions for Port Security	1501
Information About Port Security	1502
Port Security	1502
Types of Secure MAC Addresses	1502
Sticky Secure MAC Addresses	1502
Security Violations	1502
Port Security Aging	1504
Default Port Security Configuration	1504
Port Security Configuration Guidelines	1504
How to Configure Port Security	1506
Enabling and Configuring Port Security	1506
Enabling and Configuring Port Security Aging	1511

Configuration Examples for Port Security	1512
Information About Protocol Storm Protection	1513
Protocol Storm Protection	1513
Default Protocol Storm Protection Configuration	1514
How to Configure Protocol Storm Protection	1514
Enabling Protocol Storm Protection	1514
Monitoring Protocol Storm Protection	1515

CHAPTER 73
Configuring IPv6 First Hop Security 1517

Finding Feature Information	1517
Prerequisites for First Hop Security in IPv6	1517
Restrictions for First Hop Security in IPv6	1517
Information about First Hop Security in IPv6	1518
How to Configure an IPv6 Snooping Policy	1520
How to Attach an IPv6 Snooping Policy to an Interface	1522
How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface	1523
How to Attach an IPv6 Snooping Policy to VLANs Globally	1524
How to Configure the IPv6 Binding Table Content	1525
How to Configure an IPv6 Neighbor Discovery Inspection Policy	1526
How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface	1528
How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface	1529
How to Attach an IPv6 Neighbor Discovery Inspection Policy to VLANs Globally	1531
How to Configure an IPv6 Router Advertisement Guard Policy	1531
How to Attach an IPv6 Router Advertisement Guard Policy to an Interface	1534
How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface	1535
How to Attach an IPv6 Router Advertisement Guard Policy to VLANs Globally	1536
How to Configure an IPv6 DHCP Guard Policy	1536
How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface	1539
How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface	1540
How to Attach an IPv6 DHCP Guard Policy to VLANs Globally	1541
How to Configure IPv6 Source Guard	1542
How to Attach an IPv6 Source Guard Policy to an Interface	1543
How to Configure IPv6 Source Guard	1544

How to Attach an IPv6 Source Guard Policy to an Interface 1545

How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface 1546

How to Configure IPv6 Prefix Guard 1547

How to Attach an IPv6 Prefix Guard Policy to an Interface 1548

How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface 1549

CHAPTER 74 **Configuring FIPS 1551**

Information About FIPS and Common Criteria 1551

CHAPTER 75 **Configuring Control Plane Policing 1553**

Restrictions for Control Plane Policing 1553

Control Plane Policing 1553

Configuring Control Plane Policing 1554

Examples: Configuring CoPP 1555

PART X **System Management 1557**

CHAPTER 76 **Administering the System 1559**

Information About Administering the Device 1559

System Time and Date Management 1559

System Clock 1559

Real Time Clock 1560

Network Time Protocol 1560

 NTP Stratum 1561

 NTP Associations 1562

 NTP Security 1562

 NTP Implementation 1562

 NTP Version 4 1563

System Name and Prompt 1563

 Default System Name and Prompt Configuration 1563

DNS 1563

 Default DNS Settings 1564

Login Banners 1564

 Default Banner Configuration 1564

MAC Address Table	1564
MAC Address Table Creation	1565
MAC Addresses and VLANs	1565
Default MAC Address Table Settings	1565
ARP Table Management	1565
How to Administer the Device	1566
Configuring the Time and Date Manually	1566
Setting the System Clock	1566
Configuring the Time Zone	1567
Configuring Summer Time (Daylight Saving Time)	1568
Configuring a System Name	1571
Setting Up DNS	1572
Configuring a Message-of-the-Day Login Banner	1574
Configuring a Login Banner	1575
Managing the MAC Address Table	1576
Changing the Address Aging Time	1576
Configuring MAC Address Change Notification Traps	1578
Configuring MAC Address Move Notification Traps	1580
Configuring MAC Threshold Notification Traps	1582
Adding and Removing Static Address Entries	1583
Configuring Unicast MAC Address Filtering	1584
Monitoring and Maintaining Administration of the Device	1586
Configuration Examples for Device Administration	1587
Example: Setting the System Clock	1587
Examples: Configuring Summer Time	1587
Example: Configuring a MOTD Banner	1587
Example: Configuring a Login Banner	1588
Example: Configuring MAC Address Change Notification Traps	1588
Example: Configuring MAC Threshold Notification Traps	1588
Example: Adding the Static Address to the MAC Address Table	1588
Example: Configuring Unicast MAC Address Filtering	1589
CHAPTER 77	
Performing Device Setup Configuration	1591
Information About Performing Device Setup Configuration	1591

Boot Process	1591
Devices Information Assignment	1592
Default Switch Information	1592
DHCP-Based Autoconfiguration Overview	1593
DHCP Client Request Process	1593
DHCP-based Autoconfiguration and Image Update	1594
Restrictions for DHCP-based Autoconfiguration	1594
DHCP Autoconfiguration	1595
DHCP Auto-Image Update	1595
DHCP Server Configuration Guidelines	1595
Purpose of the DNS Server	1596
How to Obtain Configuration Files	1596
How to Control Environment Variables	1597
Common Environment Variables	1598
Environment Variables for TFTP	1600
Scheduled Reload of the Software Image	1600
How to Perform Device Setup Configuration	1601
Configuring DHCP Autoconfiguration (Only Configuration File)	1601
Configuring DHCP Auto-Image Update (Configuration File and Image)	1603
Configuring the Client to Download Files from DHCP Server	1606
Manually Assigning IP Information to Multiple SVIs	1607
Configuring the NVRAM Buffer Size	1609
Modifying the Device Startup Configuration	1610
Specifying the Filename to Read and Write the System Configuration	1610
Manually Booting the Switch	1611
Configuring a Scheduled Software Image Reload	1612
Monitoring Device Setup Configuration	1613
Example: Verifying the Device Running Configuration	1613
Examples: Displaying Software Install	1614
Configuration Examples for Performing Device Setup	1614
Example: Configuring a Device as a DHCP Server	1614
Example: Configuring DHCP Auto-Image Update	1615
Example: Configuring a Device to Download Configurations from a DHCP Server	1615
Example: Configuring NVRAM Buffer Size	1615

CHAPTER 78**Configuring RTU Licenses 1617**

- Finding Feature Information 1617
- Restrictions for Configuring RTU Licenses 1617
- Information About Configuring RTU Licenses 1618
 - Right-To-Use Licensing 1618
 - Right-To-Use Image-Based Licenses 1618
 - Right-To-Use License States 1618
 - Mobility Controller Mode 1619
 - Right-To-Use Adder AP-Count Rehosting Licenses 1619
- How to Configure RTU Licenses 1619
 - Activating an Image Based License 1619
 - Activating an AP-Count License 1621
 - Obtaining an Upgrade or Capacity Adder License 1621
 - Rehosting a License 1622
- Monitoring and Maintaining RTU Licenses 1622
- Configuration Examples for RTU Licensing 1623
 - Examples: Activating RTU Image Based Licenses 1623
 - Examples: Displaying RTU Licensing Information 1623
 - Example: Displaying RTU License Details 1623
 - Example: Displaying RTU License Mismatch 1623
 - Example: Displaying RTU Licensing Usage 1624

CHAPTER 79**Clustering Switches 1625**

- Understanding Switch Clusters 1625
 - Cluster Command Switch Characteristics 1626
 - Standby Cluster Command Switch Characteristics 1626
 - Candidate Switch and Cluster Member Switch Characteristics 1627
- Planning a Switch Cluster 1627
 - Automatic Discovery of Cluster Candidates and Members 1628
 - Discovery Through CDP Hops 1628
 - Discovery Through Non-CDP-Capable and Noncluster-Capable Devices 1628
 - Discovery Through Different VLANs 1629
 - Discovery Through Different Management VLANs 1630

Discovery Through Routed Ports	1631
Discovery of Newly Installed Switches	1631
HSRP and Standby Cluster Command Switches	1632
Virtual IP Addresses	1633
Other Considerations for Cluster Standby Groups	1633
Automatic Recovery of Cluster Configuration	1634
IP Addresses	1635
Hostnames	1635
Passwords	1635
SNMP Community Strings	1636
TACACS+ and RADIUS	1636
LRE Profiles	1636
Using the CLI to Manage Switch Clusters	1636
Using SNMP to Manage Switch Clusters	1637

CHAPTER 80

Configuring AVC with DNS-AS	1639
Prerequisites for AVC with DNS-AS	1639
Restrictions and Guidelines for AVC with DNS-AS	1639
Information About AVC with DNS-AS	1640
Overview of AVC with DNS-AS	1640
Key Concepts for AVC with DNS-AS	1641
AVC with DNS-AS Process Flow	1642
DNS Snooping Process	1642
DNS-AS Client Process	1642
Figure: AVC with DNS-AS Process Flow	1643
Default Configuration for AVC with DNS-AS	1644
How to Configure AVC with DNS-AS	1644
Generating Metadata Streams	1644
Configuring a DNS Server as the Authoritative Server	1646
Enabling AVC with DNS-AS	1647
Maintaining the List of Trusted Domains	1647
Configuring QoS for AVC with DNS-AS	1648
Configuring FNF for AVC with DNS-AS	1651
Option Templates	1652

Sample FNF Configuration for AVC with DNS-AS	1654
Monitoring AVC with DNS-AS	1657
Troubleshooting AVC with DNS-AS	1660
Feature History and Information for AVC with DNS-AS	1661

CHAPTER 81**Configuring SDM Templates 1663**

Finding Feature Information	1663
Information About Configuring SDM Templates	1663
Restrictions for SDM Templates	1663
SDM Templates	1663
Default Templates for Catalyst 2960-CX	1664
Default Templates for Catalyst 3560-CX	1665
How to Configure SDM Templates	1665
Setting the SDM Template	1665
Configuration Examples for SDM Templates	1667
Examples: Displaying SDM Templates	1667
Examples: Configuring SDM Templates	1667

CHAPTER 82**Configuring System Message Logs 1669**

Restrictions for Configuring System Message Logs	1669
Information About Configuring System Message Logs	1669
System Message Logging	1669
System Log Message Format	1670
Default System Message Logging Settings	1671
Enabling Syslog Trap Messages	1671
How to Configure System Message Logs	1672
Setting the Message Display Destination Device	1672
Synchronizing Log Messages	1673
Disabling Message Logging	1675
Enabling and Disabling Time Stamps on Log Messages	1676
Enabling and Disabling Sequence Numbers in Log Messages	1676
Defining the Message Severity Level	1677
Limiting Syslog Messages Sent to the History Table and to SNMP	1678
Logging Messages to a UNIX Syslog Daemon	1679

Monitoring and Maintaining System Message Logs 1680
 Monitoring Configuration Archive Logs 1680
 Configuration Examples for System Message Logs 1680
 Example: Switch System Message 1680

CHAPTER 83

Configuring Online Diagnostics 1681

Information About Configuring Online Diagnostics 1681
 Online Diagnostics 1681
 How to Configure Online Diagnostics 1682
 Starting Online Diagnostic Tests 1682
 Configuring Online Diagnostics 1682
 Scheduling Online Diagnostics 1682
 Configuring Health-Monitoring Diagnostics 1683
 Monitoring and Maintaining Online Diagnostics 1686
 Displaying Online Diagnostic Tests and Test Results 1686
 Configuration Examples for Online Diagnostic Tests 1687
 Starting Online Diagnostic Tests 1687
 Example: Configure a Health Monitoring Test 1687
 Examples: Schedule Diagnostic Test 1687
 Displaying Online Diagnostics: Examples 1688

CHAPTER 84

Data Sanitization 1691

Data Sanitization 1691
 Example: Data Sanitization 1692

CHAPTER 85

Troubleshooting the Software Configuration 1695

Information About Troubleshooting the Software Configuration 1695
 Software Failure on a Switch 1695
 Lost or Forgotten Password on a Device 1695
 Power over Ethernet Ports 1696
 Disabled Port Caused by Power Loss 1696
 Disabled Port Caused by False Link-Up 1696
 Ping 1697
 Layer 2 Traceroute 1697

Layer 2 Traceroute Guidelines	1697
IP Traceroute	1698
Time Domain Reflector Guidelines	1699
Debug Commands	1700
Onboard Failure Logging on the Switch	1700
Possible Symptoms of High CPU Utilization	1700
How to Troubleshoot the Software Configuration	1701
Recovering from a Software Failure	1701
Recovering from a Lost or Forgotten Password	1703
Procedure with Password Recovery Enabled	1704
Procedure with Password Recovery Disabled	1706
Recovering from a Command Switch Failure	1707
Replacing a Failed Command Switch with a Cluster Member	1708
Replacing a Failed Command Switch with Another Switch	1710
Preventing Autonegotiation Mismatches	1711
Troubleshooting SFP Module Security and Identification	1711
Monitoring SFP Module Status	1712
Executing Ping	1712
Monitoring Temperature	1712
Monitoring the Physical Path	1713
Executing IP Traceroute	1713
Running TDR and Displaying the Results	1713
Redirecting Debug and Error Message Output	1713
Using the show platform forward Command	1714
Configuring OBFL	1714
Verifying Troubleshooting of the Software Configuration	1715
Displaying OBFL Information	1715
Example: Verifying the Problem and Cause for High CPU Utilization	1716
Scenarios for Troubleshooting the Software Configuration	1718
Scenarios to Troubleshoot Power over Ethernet (PoE)	1718
Configuration Examples for Troubleshooting Software	1720
Example: Pinging an IP Host	1720
Example: Performing a Traceroute to an IP Host	1721
Example: Enabling All System Diagnostics	1722

CHAPTER 86 **Information About Licensing** **1723**

- Restrictions for Configuring Licenses **1723**
- Information About Licensing **1723**
 - Overview of License Levels **1723**
 - Base Licenses **1724**
 - Add-On Licenses **1724**
 - License States **1724**
 - Guidelines for License Types **1725**
 - Ordering with Smart Accounts **1725**
 - License Activation for Switch Stacks **1726**
- How to Configure Add-On License Levels **1726**
 - Activating an Image Based Add-on License **1726**
 - Activating an Image Based Add-on License on Cisco Catalyst 3560-CX Series **1727**
 - Rehosting a License **1728**
 - Monitoring Licenses **1729**
- Configuration Examples for License Levels **1729**
 - Reference **1729**
 - Example: Displaying the detailed license information **1729**
 - Example: Displaying a summary of the license information **1730**
 - Example: Displaying the end user license agreement **1730**
- Feature History for Information About Licensing **1731**

PART XI **Embedded Event Manager** **1733**

CHAPTER 87 **Embedded Event Manager Overview** **1735**

- Information About Embedded Event Manager **1735**
 - Embedded Event Manager **1735**
 - Embedded Event Manager 1.0 **1736**
 - Embedded Event Manager 2.0 **1737**
 - Embedded Event Manager 2.1 **1737**
 - Embedded Event Manager 2.1 (Software Modularity) **1738**
 - Embedded Event Manager 2.2 **1738**
 - Embedded Event Manager 2.3 **1739**

Embedded Event Manager 2.4	1739
Embedded Event Manager 3.0	1740
Embedded Event Manager 3.1	1741
Embedded Event Manager 3.2	1741
Embedded Event Manager 4.0	1742
EEM Event Detectors Available by Cisco IOS Release	1743
Event Detectors	1745
EEM Actions Available by Cisco IOS Release	1749
Embedded Event Manager Actions	1750
Embedded Event Manager Environment Variables	1750
Embedded Event Manager Policy Creation	1752
Where to Go Next	1753
Feature Information for Embedded Event Manager 4.0 Overview	1753
Additional References	1753

CHAPTER 88

Information About Writing EEM Policies Using the Cisco IOS CLI	1757
Prerequisites for Writing EEM Policies Using the Cisco IOS CLI	1757
Information About Writing EEM Policies Using the Cisco IOS CLI	1757
Embedded Event Manager Policies	1757
EEM Applet	1758
EEM Script	1758
Embedded Event Manager Built-In Environment Variables Used in EEM Applets	1758
How to Write EEM Policies Using the Cisco IOS CLI	1769
Registering and Defining an Embedded Event Manager Applet	1769
EEM Environment Variables	1769
Alphabetical Order of EEM Action Labels	1770
Troubleshooting Tips	1773
Registering and Defining an EEM Tel Script	1773
Unregistering Embedded Event Manager Policies	1775
Suspending All Embedded Event Manager Policy Execution	1776
Displaying Embedded Event Manager History Data	1777
Displaying Embedded Event Manager Registered Policies	1779
Configuring Event SNMP Notification	1780
Configuring Multiple Event Support	1781

Setting the Event Configuration Parameters	1781
Configuring EEM Class-Based Scheduling	1783
Holding a Scheduled EEM Policy Event or Event Queue	1784
Resuming Execution of EEM Policy Events or Event Queues	1785
Clearing Pending EEM Policy Events or Event Queues	1787
Modifying the Scheduling Parameters of EEM Policy Events or Event Queues	1788
Verifying Class-Based Active EEM Policies	1789
Verifying Class-Based Active EEM Policies	1790
Verifying Pending EEM Policies	1790
Configuring EEM Applet (Interactive CLI) Support	1791
Reading and Writing Input from the Active Console for Synchronous EEM Applets	1791
Configuring SNMP Library Extensions	1794
Prerequisites	1794
SNMP Get and Set Operations	1795
SNMP Traps and Inform Requests	1797
Configuring EEM Applet for SNMP Get and Set Operations	1797
Configuring EEM Applet for SNMP OID Notifications	1799
Configuring Variable Logic for EEM Applets	1802
Prerequisites	1802
Configuring Variable Logic for EEM Applets	1802
Specifying a Loop of Conditional Blocks	1802
Specifying if else Conditional Blocks	1804
Specifying foreach Iterating Statements	1805
Using Regular Expressions	1807
Incrementing the Values of Variables	1807
Configuring Event SNMP Object	1808
Disabling AAA Authorization	1810
Configuring Description of an Embedded Event Manager Applet	1811
Configuration Examples for Writing Embedded Event Manager Policies Using Tcl	1812
Embedded Event Manager Applet Configuration Examples	1812
Configuration Examples for Embedded Event Manager Applet	1817
Example Identity Event Detector	1817
Example MAT Event Detector	1817
Example Neighbor-Discovery Event Detector	1817

Embedded Event Manager Manual Policy Execution Examples	1817
Embedded Event Manager Watchdog System Monitor (Cisco IOS) Event Detector Configuration Example	1818
Configuration SNMP Library Extensions Examples	1819
SNMP Get Operations Examples	1819
SNMP GetID Operations Examples	1820
Set Operations Examples	1821
Generating SNMP Notifications Examples	1821
Configuring Variable Logic for EEM Applets Examples	1823
Configuring Event SNMP-Object Examples	1826
Configuring Description of an EEM Applet Examples	1827
Additional References	1827
Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI	1828

CHAPTER 89
Writing Embedded Event Manager Policies Using Tcl 1829

Prerequisites for Writing Embedded Event Manager Policies Using Tcl	1829
Information About Writing Embedded Event Manager Policies Using Tcl	1830
EEM Policies	1830
EEM Policy Tcl Command Extension Categories	1831
General Flow of EEM Event Detection and Recovery	1832
Safe-Tcl	1832
Bytecode Support for EEM 2.4	1834
Registration Substitution	1834
Cisco File Naming Convention for EEM	1835
How to Write Embedded Event Manager Policies Using Tcl	1836
Registering and Defining an EEM Tcl Script	1836
Displaying EEM Registered Policies	1838
Unregistering EEM Policies	1839
Suspending EEM Policy Execution	1841
Managing EEM Policies	1842
Modifying History Table Size and Displaying EEM History Data	1844
Displaying Software Modularity Process Reliability Metrics Using EEM	1845
Troubleshooting Tips	1846
Modifying the Sample EEM Policies	1847

Sample EEM Policies	1847
Programming EEM Policies with Tcl	1849
Tcl Policy Structure and Requirements	1849
EEM Entry Status	1851
EEM Exit Status	1851
EEM Policies and Cisco Error Number	1852
Troubleshooting Tips	1858
Creating an EEM User Tcl Library Index	1859
Creating an EEM User Tcl Package Index	1862
Configuration Examples for Writing Embedded Event Manager Policies Using Tcl	1865
Assigning a Username for a Tcl Session Examples	1865
EEM Event Detector Demo Examples	1865
Programming Policies with Tcl Sample Scripts Example	1873
Debugging Embedded Event Manager Policies Examples	1882
Tracing Tcl set Command Operations Example	1884
RPC Event Detector Example	1884
Additional References	1886
Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI	1887

CHAPTER 90
Signed Tcl Scripts 1889

Prerequisites for Signed Tcl Scripts	1889
Restrictions for Signed Tcl Scripts	1889
Information About Signed Tcl Scripts	1890
Cisco PKI	1890
RSA Key Pair	1890
Certificate and Trustpoint	1891
How to Configure Signed Tcl Scripts	1891
Generating a Key Pair	1891
Generating a Certificate	1892
Signing the Tcl Scripts	1894
Verifying the Signature	1894
Converting the Signature into Nonbinary Data	1895
Configuring the Device with a Certificate	1898
Verifying the Trustpoint	1902

Verifying the Signed Tcl Script	1902
What to Do Next	1903
Configuration Examples for Signed Tcl Script	1904
Generating a Key Pair Example	1904
Generating a Certificate Example	1904
Signing the Tcl Scripts Example	1905
Verifying the Signature Example	1905
Converting the Signature with Nonbinary Data Example	1905
Configuring the Device with a Certificate Example	1907
Additional References	1908
Feature Information for Signed Tcl Scripts	1909
Glossary	1909
Notices	1910
OpenSSL Open SSL Project	1910
License Issues	1910

CHAPTER 91 **EEM CLI Library Command Extensions** **1913**

cli_close	1914
cli_exec	1914
cli_get_ttyname	1915
cli_open	1915
cli_read	1916
cli_read_drain	1916
cli_read_line	1917
cli_read_pattern	1917
cli_run	1918
cli_run_interactive	1919
cli_write	1920
EEM 4.0 CLI Library XML-PI Support	1923
EEM CLI Library XML-PI Support	1923

CHAPTER 92 **EEM Context Library Command Extensions** **1925**

context_retrieve	1925
context_save	1928

CHAPTER 93 **EEM Event Registration Tcl Command Extensions** **1933**

event_register_appl	1934
event_register_cli	1936
event_register_counter	1939
event_register_gold	1941
event_register_identity	1947
event_register_interface	1949
event_register_ioswdsysmon	1954
event_register_ipsla	1957
event_register_mat	1960
event_register_neighbor_discovery	1962
event_register_nf	1965
event_register_none	1968
event_register_oir	1970
event_register_process	1972
event_register_resource	1974
event_register_rf	1976
event_register_routing	1979
event_register_rpc	1981
event_register_snmp	1983
event_register_snmp_notification	1987
event_register_snmp_object	1989
event_register_syslog	1992
event_register_timer	1994
event_register_timer_subscriber	1998
event_register_track	2000
event_register_wdsysmon	2002

CHAPTER 94 **EEM Event Tcl Command Extensions** **2017**

event_completion	2017
event_completion_with_wait	2018
event_publish	2019
event_wait	2022

CHAPTER 95	EEM Library Debug Command Extensions	2025
	cli_debug	2025
	smtp_debug	2025

CHAPTER 96	EEM Multiple Event Support Tcl Command Extensions	2027
	attribute	2027
	correlate	2028
	trigger	2029

CHAPTER 97	EEM SMTP Library Command Extensions	2031
	smtp_send_email	2032
	smtp_subst	2033

CHAPTER 98	EEM System Information Tcl Command Extensions	2035
	sys_reqinfo_cli_freq	2036
	sys_reqinfo_cli_history	2037
	sys_reqinfo_cpu_all	2037
	sys_reqinfo_crash_history	2038
	sys_reqinfo_mem_all	2039
	sys_reqinfo_proc	2040
	sys_reqinfo_proc_all	2042
	sys_reqinfo_routename	2042
	sys_reqinfo_snmp	2043
	sys_reqinfo_syslog_freq	2044
	sys_reqinfo_syslog_history	2045

CHAPTER 99	EEM Utility Tcl Command Extensions	2047
	appl_read	2048
	appl_reqinfo	2048
	appl_setinfo	2049
	counter_modify	2050
	description	2051

fts_get_stamp 2052
 register_counter 2053
 register_timer 2054
 timer_arm 2056
 timer_cancel 2057
 unregister_counter 2058

PART XII

VLAN 2061

CHAPTER 100

Configuring VTP 2063

Finding Feature Information 2063
 Prerequisites for VTP 2063
 Restrictions for VTP 2064
 Information About VTP 2064
 VTP 2064
 VTP Domain 2064
 VTP Modes 2065
 VTP Advertisements 2066
 VTP Version 2 2067
 VTP Version 3 2067
 VTP Pruning 2068
 VTP Configuration Guidelines 2068
 VTP Configuration Requirements 2068
 VTP Settings 2068
 Domain Names for Configuring VTP 2069
 Passwords for the VTP Domain 2069
 VTP Version 2069
 Default VTP Configuration 2070
 How to Configure VTP 2071
 Configuring VTP Mode 2071
 Configuring a VTP Version 3 Password 2073
 Configuring a VTP Version 3 Primary Server 2074
 Enabling the VTP Version 2075
 Enabling VTP Pruning 2076

Configuring VTP on a Per-Port Basis	2078
Adding a VTP Client to a VTP Domain	2079
Monitoring VTP	2081
Configuration Examples for VTP	2081
Example: Configuring a Switch as the Primary Server	2081
Example: Configuring Switch as VTP Server	2082
Example: Enabling VTP on the Interface	2082
Example: Creating the VTP Password	2082
Where to Go Next	2082

CHAPTER 101**Configuring VLANs 2085**

Finding Feature Information	2085
Prerequisites for VLANs	2085
Restrictions for VLANs	2085
Information About VLANs	2086
Logical Networks	2086
Supported VLANs	2086
VLAN Port Membership Modes	2086
VLAN Configuration Files	2087
Normal-Range VLAN Configuration Guidelines	2088
Extended-Range VLAN Configuration Guidelines	2089
Default VLAN Configurations	2089
Default Ethernet VLAN Configuration	2089
How to Configure VLANs	2090
How to Configure Normal-Range VLANs	2090
Creating or Modifying an Ethernet VLAN	2091
Deleting a VLAN	2093
Assigning Static-Access Ports to a VLAN	2094
How to Configure Extended-Range VLANs	2095
Creating an Extended-Range VLAN	2096
Monitoring VLANs	2097
Configuration Examples	2098
Example: Creating a VLAN Name	2098
Example: Configuring a Port as Access Port	2098

Example: Creating an Extended-Range VLAN	2098
Where to Go Next	2098

CHAPTER 102

Configuring VLAN Trunks	2099
Finding Feature Information	2099
Prerequisites for VLAN Trunks	2099
Information About VLAN Trunks	2100
Trunking Overview	2100
Trunking Modes	2100
Layer 2 Interface Modes	2100
Allowed VLANs on a Trunk	2101
Load Sharing on Trunk Ports	2101
Network Load Sharing Using STP Priorities	2101
Network Load Sharing Using STP Path Cost	2102
Feature Interactions	2102
Default Layer 2 Ethernet Interface VLAN Configuration	2102
How to Configure VLAN Trunks	2103
Configuring an Ethernet Interface as a Trunk Port	2103
Configuring a Trunk Port	2103
Defining the Allowed VLANs on a Trunk	2105
Changing the Pruning-Eligible List	2106
Configuring the Native VLAN for Untagged Traffic	2108
Configuring Trunk Ports for Load Sharing	2109
Configuring Load Sharing Using STP Port Priorities	2109
Configuring Load Sharing Using STP Path Cost	2113
Configuration Examples for VLAN Trunking	2116
Example: Configuring a Trunk Port	2116
Example: Removing a VLAN from a Port	2116
Where to Go Next	2116

CHAPTER 103

Configuring VMPS	2117
Finding Feature Information	2117
Prerequisites for VMPS	2117
Restrictions for VMPS	2117

Information About VMPS	2118
Dynamic VLAN Assignments	2118
Dynamic-Access Port VLAN Membership	2119
Default VMPS Client Configuration	2119
How to Configure VMPS	2120
Entering the IP Address of the VMPS	2120
Configuring Dynamic-Access Ports on VMPS Clients	2121
Reconfirming VLAN Memberships	2123
Changing the Reconfirmation Interval	2123
Changing the Retry Count	2125
Troubleshooting Dynamic-Access Port VLAN Membership	2126
Monitoring the VMPS	2126
Configuration Example for VMPS	2127
Example: VMPS Configuration	2127
Where to Go Next	2128

CHAPTER 104

Configuring Voice VLANs	2129
Finding Feature Information	2129
Prerequisites for Voice VLANs	2129
Restrictions for Voice VLANs	2130
Information About Voice VLAN	2130
Voice VLANs	2130
Cisco IP Phone Voice Traffic	2130
Cisco IP Phone Data Traffic	2130
Voice VLAN Configuration Guidelines	2131
Default Voice VLAN Configuration	2132
How to Configure Voice VLAN	2132
Configuring Cisco IP Phone Voice Traffic	2132
Configuring the Priority of Incoming Data Frames	2134
Monitoring Voice VLAN	2136
Configuration Examples	2136
Example: Configuring Cisco IP Phone Voice Traffic	2136
Example: Configuring the Priority of Incoming Data Frames	2136
Where to Go Next	2136

CHAPTER 105**Configuring Private VLANs 2137**

- Finding Feature Information 2137
- Prerequisites for Private VLANs 2137
- Restrictions for Private VLANs 2137
- Information About Private VLANs 2139
 - Private VLAN Domains 2139
 - Secondary VLANs 2139
 - Private VLANs Ports 2139
 - Private VLANs in Networks 2140
 - IP Addressing Scheme with Private VLANs 2141
 - Private VLANs Across Multiple Devices 2141
 - Private-VLAN Interaction with Other Features 2142
 - Private VLANs and Unicast, Broadcast, and Multicast Traffic 2142
 - Private VLANs and SVIs 2142
 - Private-VLAN Configuration Guidelines 2143
 - Secondary and Primary VLAN Configuration 2143
 - Private VLAN Port Configuration 2145
 - Private VLAN Configuration Tasks 2145
- How to Configure Private VLANs 2145
 - Configuring and Associating VLANs in a Private VLAN 2145
 - Configuring a Layer 2 Interface as a Private VLAN Host Port 2149
 - Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port 2151
 - Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface 2152
- Monitoring Private VLANs 2154
- Configuration Examples for Private VLANs 2154
 - Example: Configuring an Interface as a Host Port 2154
 - Example: Configuring an Interface as a Private VLAN Promiscuous Port 2155
 - Example: Mapping Secondary VLANs to a Primary VLAN Interface 2155
 - Example: Monitoring Private VLANs 2156
- Where to Go Next 2156
- Additional References 2156



Preface

This preface describes the conventions of this document and information on how to obtain other documentation. It also provides information on what's new in Cisco product documentation.

- [Document Conventions](#) , on page lxxxiii
- [Related Documentation](#), on page lxxxv
- [Obtaining Documentation and Submitting a Service Request](#), on page lxxxv

Document Conventions

This document uses the following conventions:

Convention	Description
^ or Ctrl	Both the ^ symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination ^D or Ctrl-D means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)
bold font	Commands and keywords and user-entered text appear in bold font .
<i>Italic font</i>	Document titles, new or emphasized terms, and arguments for which you supply values are in <i>italic font</i> .
Courier font	Terminal sessions and information the system displays appear in <i>courier font</i> .
Bold Courier font	Bold Courier font indicates text that the user must enter.
[x]	Elements in square brackets are optional.
...	An ellipsis (three consecutive nonbolded periods without spaces) after a syntax element indicates that the element can be repeated.
	A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.
[x y]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Convention	Description
{x y}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x {y z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
<>	Nonprinting characters such as passwords are in angle brackets.
[]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Reader Alert Conventions

This document may use the following conventions for reader alerts:



Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.



Tip Means *the following information will help you solve a problem*.



Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.



Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.



Warning IMPORTANT SAFETY INSTRUCTIONS

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device. Statement 1071

SAVE THESE INSTRUCTIONS

Related Documentation

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.



PART I

Interface and Hardware

- [Configuring Interface Characteristics, on page 1](#)
- [Configuring Auto-MDIX, on page 25](#)
- [Configuring LLDP, LLDP-MED, and Wired Location Service, on page 29](#)
- [Configuring MultiGigabit Ports on WS-C3560CX-8PD-S, on page 47](#)
- [Configuring System MTU, on page 51](#)
- [Configuring Boot Fast, on page 53](#)
- [Configuring Power over Ethernet, on page 57](#)
- [Configuring 2-event Classification, on page 73](#)
- [Configuring EEE, on page 75](#)



CHAPTER 1

Configuring Interface Characteristics

- [Information About Configuring Interface Characteristics, on page 1](#)
- [How to Configure Interface Characteristics, on page 9](#)
- [Monitoring Interface Characteristics, on page 21](#)
- [Configuration Examples for Interface Characteristics, on page 22](#)

Information About Configuring Interface Characteristics

Interface Types

This section describes the different types of interfaces supported by the device. The rest of the chapter describes configuration procedures for physical interface characteristics.

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for normal-range VLANs (VLAN IDs 1 to 1005) are saved in the VLAN database. If VTP is version 1 or 2, to configure extended-range VLANs (VLAN IDs 1006 to 4094), you must first set VTP mode to transparent. Extended-range VLANs created in transparent mode are not added to the VLAN database but are saved in the device running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode. These VLANs are saved in the VLAN database.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and, if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.

Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port or a trunk port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to set the switchport mode by negotiating with the port on the other end of the link. Switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands.

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives a tagged packet (Inter-Switch Link [ISL] or IEEE 802.1Q tagged), the packet is dropped, and the source address is not learned.

The types of access ports supported are:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x).
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the device are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the device cannot be a VMPS server.

You can also configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database.

The device supports only IEEE 802.1Q trunk ports. An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An IEEE 802.1Q trunk port is assigned a default port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default, a trunk port is a member of every VLAN known to the VTP, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if VTP knows of the VLAN and if the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. You can associate only one SVI with a VLAN. You configure an SVI for a VLAN

only to route between VLANs or to provide IP host connectivity to the device. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote device administration. Additional SVIs must be explicitly configured.



Note You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system. SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an ISL or IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address.

You can also use the interface range command to configure existing VLAN SVIs within the range. The commands entered under the interface range command are applied to all existing VLAN SVIs within the range. You can enter the command **interface range create vlan** *x - y* to create all VLANs in the specified range that do not already exist. When the VLAN interface is created, **interface range vlan** *id* can be used to configure the VLAN interface.

When you create an SVI, it does not become active until it is associated with a physical port.

SVI Autostate Exclude

The line state of an SVI with multiple ports on a VLAN is in the *up* state when it meets these conditions:

- The VLAN exists and is active in the VLAN database on the device
- The VLAN interface exists and is not administratively down.
- At least one Layer 2 (access or trunk) port exists, has a link in the *up* state on this VLAN, and is in the spanning-tree forwarding state on the VLAN.



Note The protocol link state for VLAN interfaces come up when the first switchport belonging to the corresponding VLAN link comes up and is in STP forwarding state.

The default action, when a VLAN has multiple ports, is that the SVI goes down when all ports in the VLAN go down. You can use the SVI autostate exclude feature to configure a port so that it is not included in the SVI line-state up-or-down calculation. For example, if the only active port on the VLAN is a monitoring port, you might configure autostate exclude on that port so that the VLAN goes down when all other ports go down. When enabled on a port, **autostate exclude** applies to all VLANs that are enabled on that port.

The VLAN interface is brought up when one Layer 2 port in the VLAN has had time to converge (transition from STP listening-learning state to forwarding state). This prevents features such as routing protocols from using the VLAN interface as if it were fully operational and minimizes other problems.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between devices or between devices and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical

trunk port or multiple access ports into one logical access port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together.



Note Cisco Catalyst 2960-CX and 3560-CX support a maximum of six EtherChannel port groups.

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the device senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

Using the Switch USB Ports

USB Mini-Type B Console Port

The device has the following console ports:

- USB mini-Type B console connection
- RJ-45 console port

Console output appears on devices connected to both ports, but console input is active on only one port at a time. By default, the USB connector takes precedence over the RJ-45 connector.



Note Windows PCs require a driver for the USB port. See the hardware installation guide for driver installation instructions.

Use the supplied USB Type A-to-USB mini-Type B cable to connect a PC or other device to the device. The connected device must include a terminal emulation application. When the device detects a valid USB connection to a powered-on device that supports host functionality (such as a PC), input from the RJ-45 console is immediately disabled, and input from the USB console is enabled. Removing the USB connection immediately reenables input from the RJ-45 console connection. An LED on the device shows which console connection is in use.

Console Port Change Logs

At software startup, a log shows whether the USB or the RJ-45 console is active. Every device always first displays the RJ-45 media type.

When the USB cable is removed or the PC de-activates the USB connection, the hardware automatically changes to the RJ-45 console interface:

You can configure the console type to always be RJ-45, and you can configure an inactivity timeout for the USB connector.

USB Type A Ports

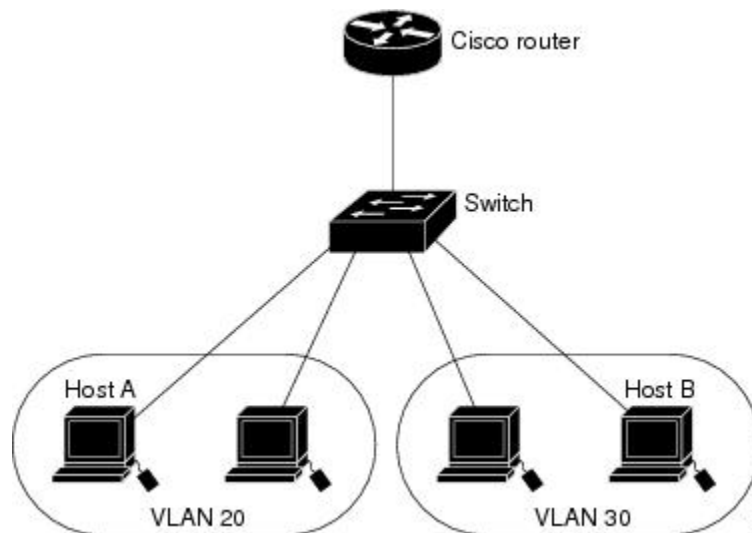
The USB Type A ports provide access to external USB flash devices, also known as thumb drives or USB keys. The switch supports Cisco 64 MB, 256 MB, 512 MB, 1 GB, 4 GB, and 8 GB flash drives. You can use standard Cisco IOS command-line interface (CLI) commands to read, write, erase, and copy to or from the flash device. You can also configure the switch to boot from the USB flash drive.

Interface Connections

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device.

In the following configuration example, when Host A in VLAN 20 sends data to Host B in VLAN 30, the data must go from Host A to the device, to the router, back to the device, and then to Host B.

Figure 1: Connecting VLANs with the Switch



With a standard Layer 2 switch, ports in different VLANs have to exchange information through a router.



Note The Catalyst 3560-CX and 2960-CX switches do not support stacking. Ignore all references to stacking throughout this book.

Interface Configuration Mode

The device supports these interface types:

- Physical ports—device ports and routed ports
- VLANs—switch virtual interfaces
- Port channels—EtherChannel interfaces

You can also configure a range of interfaces.

To configure a physical interface (port), specify the interface type, module number, and device port number, and enter interface configuration mode.

- Type—Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces (gigabitethernet or gi).
- Module number—The module or slot number on the switch (always 0).
- Port number—The interface number on the switch. The 10/100/1000 port numbers always begin at 1, starting with the far left port when facing the front of the switch, for example, gigabitethernet1/0/1 or gigabitethernet1/0/8. For a switch with 10/100/1000 ports and SFP module ports, SFP module ports are numbered consecutively following the 10/100/1000 ports.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

Default Ethernet Interface Configuration

This table shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces.

Table 1: Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (switchport command).
Allowed VLAN range	VLANs 1– 4094.
Default VLAN (for access ports)	VLAN 1.
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1.
802.1p priority-tagged traffic	Drop all packets tagged with VLAN 0.
VLAN trunking	Switchport mode dynamic auto (supports DTP).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)

Feature	Default Setting
Duplex mode	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Flow control	Flow control is set to receive: off . It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports.
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked).
Broadcast, multicast, and unicast storm control	Disabled.
Protected port	Disabled.
Port security	Disabled.
Port Fast	Disabled.
Auto-MDIX	Enabled. Note The device might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the device through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.
Power over Ethernet (PoE)	Enabled (auto).
Keepalive messages	Disabled on SFP module ports; enabled on all other ports.

Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, or 1000 Mb/s and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mb/s ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch modules include Gigabit Ethernet (10/100/1000-Mb/s) ports and small form-factor pluggable (SFP) module slots supporting SFP modules.

Speed and Duplex Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- Do not disable Auto-Negotiation on PoE switches.
- Gigabit Ethernet (10/100/1000-Mb/s) ports support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s do not support half-duplex mode.

- For SFP module ports, the speed and duplex CLI options change depending on the SFP module type:
 - The 1000BASE-*x* (where *x* is -BX, -CWDM, -LX, -SX, and -ZX) SFP module ports support the **nonnegotiate** keyword in the **speed** interface configuration command. Duplex options are not supported.
 - The 1000BASE-T SFP module ports support the same speed and duplex options as the 10/100/1000-Mb/s ports.
 -
- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- When STP is enabled and a port is reconfigured, the device can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.
- As best practice, we suggest configuring the speed and duplex options on a link to auto or to fixed on both the ends. If one side of the link is configured to auto and the other side is configured to fixed, the link will not be up and this is expected.



Caution Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

IEEE 802.3x Flow Control

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port by sending a pause frame to stop sending until the condition clears. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.



Note The switch ports can receive, but not send, pause frames.

Use the **flowcontrol** interface configuration command to set the interface's ability to **receive** pause frames to **on**, **off**, or **desired**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to flow control settings on the device:

- **receive on** (or **desired**): The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off**: Flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.



Note For details on the command settings and the resulting flow control resolution on local and remote ports, see the **flowcontrol** interface configuration command in the command reference for this release.

How to Configure Interface Characteristics

Configuring Interfaces

These general instructions apply to all interface configuration processes.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface Example: <pre>Switch(config)# interface gigabitethernet 1/0/1 Switch(config-if)#</pre>	Identifies the interface type and the number of the connector. <p>Note You do not need to add a space between the interface type and the interface number. For example, in the preceding line, you can specify either gigabitethernet 1/0/1, gigabitethernet1/0/1, gi 1/0/1, or gi1/0/1.</p>
Step 4	Follow each interface command with the interface configuration commands that the interface requires.	Defines the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter end to return to privileged EXEC mode.
Step 5	interface range or interface range macro	(Optional) Configures a range of interfaces. <p>Note Interfaces configured in a range must be the same type and must be configured with the same feature options.</p>
Step 6	show interfaces	Displays a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Adding a Description for an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **description** *string*
5. **end**
6. **show interfaces** *interface-id* **description**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2	Specifies the interface for which you are adding a description, and enter interface configuration mode.
Step 4	description <i>string</i> Example: Switch(config-if)# description Connects to Marketing	Adds a description (up to 240 characters) for an interface.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> description	Verifies your entry.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Range of Interfaces

To configure multiple interfaces with the same configuration parameters, use the **interface range** global configuration command. When you enter the interface-range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface range** {*port-range* | **macro** *macro_name*}
4. **end**
5. **show interfaces** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface range { <i>port-range</i> macro <i>macro_name</i> } Example: <pre>Switch(config)# interface range macro</pre>	Specifies the range of interfaces (VLANs or physical ports) to be configured, and enter interface-range configuration mode. <ul style="list-style-type: none"> • You can use the interface range command to configure up to five port ranges or a previously defined macro. • The macro variable is explained in the section on <i>Configuring and Using Interface Range Macros</i>.

	Command or Action	Purpose
		<ul style="list-style-type: none"> In a comma-separated <i>port-range</i>, you must enter the interface type for each entry and enter spaces before and after the comma. In a hyphen-separated <i>port-range</i>, you do not need to re-enter the interface type, but you must enter a space before the hyphen. <p>Note Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.</p>
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show interfaces [<i>interface-id</i>] Example: <pre>Switch# show interfaces</pre>	Verifies the configuration of the interfaces in the range.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **define interface-range** *macro_name* *interface-range*
4. **interface range macro** *macro_name*
5. **end**
6. **show running-config | include define**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	define interface-range <i>macro_name interface-range</i> Example: <pre>Switch(config)# define interface-range enet_list gigabitethernet 1/0/1 - 2</pre>	Defines the interface-range macro, and save it in NVRAM. <ul style="list-style-type: none"> • The <i>macro_name</i> is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each <i>interface-range</i> must consist of the same port type. <p>Note Before you can use the macro keyword in the interface range macro global configuration command string, you must use the define interface-range global configuration command to define the macro.</p>
Step 4	interface range macro <i>macro_name</i> Example: <pre>Switch(config)# interface range macro enet_list</pre>	Selects the interface range to be configured using the values saved in the interface-range macro called <i>macro_name</i> . You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config include define Example: <pre>Switch# show running-config include define</pre>	Shows the defined interface range macro configuration.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Ethernet Interfaces

Setting the Interface Speed and Duplex Parameters

SUMMARY STEPS

1. enable
2. configure terminal
3. interface *interface-id*
4. duplex {auto | full | half}
5. end
6. show interfaces *interface-id*
7. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/3</pre>	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 4	duplex {auto full half} Example:	This command is not available on a 10-Gigabit Ethernet interface. Enter the duplex parameter for the interface.

	Command or Action	Purpose
	<code>Switch(config-if)# duplex half</code>	Enable half-duplex mode (for interfaces operating only at 10 or 100 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 Mb/s. You can configure the duplex setting when the speed is set to auto .
Step 5	end Example: <code>Switch(config-if)# end</code>	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> Example: <code>Switch# show interfaces gigabitethernet 1/0/3</code>	Displays the interface speed and duplex mode configuration.
Step 7	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring IEEE 802.3x Flow Control

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **flowcontrol {receive} {on | off | desired}**
4. **end**
5. **show interfaces *interface-id***
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode
Step 2	interface <i>interface-id</i> Example:	Specifies the physical interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	flowcontrol {receive} {on off desired} Example: Switch(config-if)# flowcontrol receive on	Configures the flow control mode for the port.
Step 4	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 5	show interfaces <i>interface-id</i> Example:	Verifies the interface flow control settings.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring SVI Autostate Exclude

SUMMARY STEPS

1. enable
2. configure terminal
3. interface *interface-id*
4. switchport autostate exclude
5. end
6. show running config interface *interface-id*
7. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet1/0/2</code>	Specifies a Layer 2 interface (physical port or port channel), and enter interface configuration mode.
Step 4	switchport autostate exclude Example: Switch(config-if)# <code>switchport autostate exclude</code>	Excludes the access or trunk port when defining the status of an SVI line state (up or down)
Step 5	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config interface <i>interface-id</i>	(Optional) Shows the running configuration. Verifies the configuration.
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface {vlan vlan-id} | { gigabitethernetinterface-id} | {port-channel port-channel-number}`
4. `shutdown`
5. `no shutdown`
6. `end`
7. `show running-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface {vlan <i>vlan-id</i> } { gigabitethernet <i>interface-id</i> } { port-channel <i>port-channel-number</i> } Example: Switch(config)# interface gigabitethernet 1/0/2	Selects the interface to be configured.
Step 4	shutdown Example: Switch(config-if)# shutdown	Shuts down an interface.
Step 5	no shutdown Example: Switch(config-if)# no shutdown	Restarts an interface.
Step 6	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Switch# show running-config	Verifies your entries.

Configuring the Console Media Type

Follow these steps to set the console media type to RJ-45. If you configure the console as RJ-45, USB console operation is disabled, and input comes only through the RJ-45 connector.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **media-type rj45**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	line console 0 Example: <pre>Switch(config)# line console 0</pre>	Configures the console and enters line configuration mode.
Step 4	media-type rj45 Example: <pre>Switch(config-line)# media-type rj45</pre>	Configures the console media type to be only RJ-45 port. If you do not enter this command and both types are connected, the USB port is used by default.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Configuring the USB Inactivity Timeout

The configurable inactivity timeout reactivates the RJ-45 console port if the USB console port is activated but no input activity occurs on it for a specified time period. When the USB console port is deactivated due to a timeout, you can restore its operation by disconnecting and reconnecting the USB cable.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `line console 0`
4. `usb-inactivity-timeout timeout-minutes`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code> Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	<code>configure terminal</code> Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>line console 0</code> Example: Switch(config)# <code>line console 0</code>	Configures the console and enters line configuration mode.
Step 4	<code>usb-inactivity-timeout timeout-minutes</code> Example: Switch(config-line)# <code>usb-inactivity-timeout 30</code>	Specify an inactivity timeout for the console port. The range is 1 to 240 minutes. The default is to have no timeout configured.
Step 5	<code>copy running-config startup-config</code> Example:	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# <code>copy running-config startup-config</code>	

Monitoring Interface Characteristics

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces.

Table 2: Show Commands for Interfaces

Command	Purpose
show interfaces <i>interface-number</i> downshift module <i>module-number</i>	Displays the downshift status details of the specified interfaces and modules.
show interfaces <i>interface-id</i> status [err-disabled]	Displays interface status or a list of interfaces in the error-disabled state.
show interfaces [<i>interface-id</i>] switchport	Displays administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [<i>interface-id</i>] description	Displays the description configured on an interface or all interfaces and the interface status.
show ip interface [<i>interface-id</i>]	Displays the usability status of all interfaces configured for IP routing or the specified interface.
show interface [<i>interface-id</i>] stats	Displays the input and output packets by the switching path for the interface.
show interfaces <i>interface-id</i>	(Optional) Displays speed and duplex on the interface.
show interfaces transceiver dom-supported-list	(Optional) Displays Digital Optical Monitoring (DOM) status on the connect SFP modules.
show interfaces transceiver properties	(Optional) Displays temperature, voltage, or amount of current on the interface.
show interfaces [<i>interface-id</i>] [{ transceiver properties detail }] <i>module number</i>	Displays physical and operational status about an SFP module.
show running-config interface [<i>interface-id</i>]	Displays the running configuration in RAM for the interface.
show version	Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images.

Command	Purpose
<code>show controllers ethernet-controller interface-id phy</code>	Displays the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 3: Clear Commands for Interfaces

Command	Purpose
<code>clear counters [interface-id]</code>	Clears interface counters.
<code>clear interface interface-id</code>	Resets the hardware logic on an interface.
<code>clear line [number console 0 vty number]</code>	Resets the hardware logic on an asynchronous serial line.



Note The `clear counters` privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the `show interface` privileged EXEC command.

Configuration Examples for Interface Characteristics

Configuring a Range of Interfaces: Examples

This example shows how to use the `interface range` global configuration command to set the speed to 100 Mb/s on ports 1 to 4 on switch 1:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet 1/0/1 - 4
Switch(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Gigabit Ethernet ports 1 to 3 and 10-Gigabit Ethernet ports 1 and 2 to receive flow-control pause frames:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/1/1 - 2
Switch(config-if-range)# flowcontrol receive on
```

If you enter multiple configuration commands while you are in interface-range mode, each command is executed as it is entered. The commands are not batched and executed after you exit interface-range mode. If you exit interface-range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface-range configuration mode.

Configuring and Using Interface Range Macros: Examples

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 on switch 1 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# define interface-range enet_list gigabitethernet 1/1/1 - 2
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list gigabitethernet 1/1/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 gigabitethernet1/1/1 - 2, gigabitethernet1/1/5
- 7, tengigabitethernet1/1/1 -2
Switch(config)# end
```

This example shows how to enter interface-range configuration mode for the interface-range macro *enet_list*:

```
Switch# configure terminal
Switch(config)# interface range macro enet_list
Switch(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Switch# configure terminal
Switch(config)# no define interface-range enet_list
Switch(config)# end
Switch# show run | include define
Switch#
```

Setting Interface Speed and Duplex Mode: Example

This example shows how to set the interface speed to 100 Mb/s and the duplex mode to half on a 10/100/1000 Mb/s port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet 1/0/3
Switch(config-if)# speed 100
Switch(config-if)# duplex half
```

This example shows how to set the interface speed to 100 Mb/s on a 10/100/1000 Mb/s port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet 1/0/2
Switch(config-if)# speed 100
```

Configuring the Console Media Type: Example

This example disables the USB console media type and enables the RJ-45 console media type.

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# media-type rj45
```

This example reverses the previous configuration and immediately activates any USB console that is connected.

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# no media-type rj45
```

Configuring the USB Inactivity Timeout: Example

This example configures the inactivity timeout to 30 minutes:

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# usb-inactivity-timeout 30
```

To disable the configuration, use these commands:

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# no usb-inactivity-timeout
```

If there is no (input) activity on a USB console port for the configured number of minutes, the inactivity timeout setting applies to the RJ-45 port, and a log shows this occurrence:

```
*Mar  1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled
due to inactivity, media-type reverted to RJ45.
```

At this point, the only way to reactivate the USB console port is to disconnect and reconnect the cable.

When the USB cable on the switch has been disconnected and reconnected, a log similar to this appears:

```
*Mar  1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```




CHAPTER 2

Configuring Auto-MDIX

- [Prerequisites for Auto-MDIX, on page 25](#)
- [Restrictions for Auto-MDIX, on page 25](#)
- [Information About Configuring Auto-MDIX, on page 25](#)
- [How to Configure Auto-MDIX, on page 26](#)
- [Example for Configuring Auto-MDIX, on page 27](#)

Prerequisites for Auto-MDIX

Automatic medium-dependent interface crossover (auto-MDIX) is enabled by default.

Auto-MDIX is supported on all 10/100/1000-Mb/s and on 10/100/1000BASE-TX small form-factor pluggable (SFP)-module interfaces. It is not supported on 1000BASE-SX or -LX SFP module interfaces.

Restrictions for Auto-MDIX

The device might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the device through a crossover cable. This is regardless of whether auto-MDIX is enabled on the switch port.

Information About Configuring Auto-MDIX

Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting devices without the auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other devices or repeaters. With auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, see the hardware installation guide.

This table shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 4: Link Conditions and Auto-MDIX Settings

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

How to Configure Auto-MDIX

Configuring Auto-MDIX on an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **speed auto**
5. **duplex auto**
6. **end**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode
Step 3	interface <i>interface-id</i> Example:	Specifies the physical interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Switch(config)# interface gigabitethernet 1/0/1	
Step 4	speed auto Example: Switch(config-if)# speed auto	Configures the interface to autonegotiate speed with the connected device.
Step 5	duplex auto Example: Switch(config-if)# duplex auto	Configures the interface to autonegotiate duplex mode with the connected device.
Step 6	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Example for Configuring Auto-MDIX

This example shows how to enable auto-MDIX on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```




CHAPTER 3

Configuring LLDP, LLDP-MED, and Wired Location Service

- [Information About LLDP, LLDP-MED, and Wired Location Service, on page 29](#)
- [How to Configure LLDP, LLDP-MED, and Wired Location Service, on page 33](#)
- [Configuration Examples for LLDP, LLDP-MED, and Wired Location Service, on page 44](#)
- [Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service, on page 45](#)

Information About LLDP, LLDP-MED, and Wired Location Service

LLDP

The Cisco Discovery Protocol (CDP) is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, switches, and controllers). CDP allows network management applications to automatically discover and learn about other Cisco devices connected to the network.

To support non-Cisco devices and to allow for interoperability between other devices, the device supports the IEEE 802.1AB Link Layer Discovery Protocol (LLDP). LLDP is a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the data-link layer, which allows two systems running different network layer protocols to learn about each other.

LLDP Supported TLVs

LLDP supports a set of attributes that it uses to discover neighbor devices. These attributes contain type, length, and value descriptions and are referred to as TLVs. LLDP supported devices can use TLVs to receive and send information to their neighbors. This protocol can advertise details such as configuration information, device capabilities, and device identity.

The switch supports these basic management TLVs. These are mandatory LLDP TLVs.

- Port description TLV
- System name TLV
- System description TLV

- System capabilities TLV
- Management address TLV

These organizationally specific LLDP TLVs are also advertised to support LLDP-MED.

- Port VLAN ID TLV (IEEE 802.1 organizationally specific TLVs)
- MAC/PHY configuration/status TLV (IEEE 802.3 organizationally specific TLVs)

LLDP and Cisco Medianet

When you configure LLDP or CDP location information on a per-port basis, remote devices can send Cisco Medianet location information to the device.

LLDP-MED

LLDP for Media Endpoint Devices (LLDP-MED) is an extension to LLDP that operates between endpoint devices such as IP phones and network devices. It specifically provides support for voice over IP (VoIP) applications and provides additional TLVs for capabilities discovery, network policy, Power over Ethernet, inventory management and location information. By default, all LLDP-MED TLVs are enabled.

LLDP-MED Supported TLVs

LLDP-MED supports these TLVs:

- LLDP-MED capabilities TLV

Allows LLDP-MED endpoints to determine the capabilities that the connected device supports and has enabled.

- Network policy TLV

Allows both network connectivity devices and endpoints to advertise VLAN configurations and associated Layer 2 and Layer 3 attributes for the specific application on that port. For example, the switch can notify a phone of the VLAN number that it should use. The phone can connect to any device, obtain its VLAN number, and then start communicating with the call control.

By defining a network-policy profile TLV, you can create a profile for voice and voice-signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode. These profile attributes are then maintained centrally on the switch and propagated to the phone.

- Power management TLV

Enables advanced power management between LLDP-MED endpoint and network connectivity devices. Allows devices and phones to convey power information, such as how the device is powered, power priority, and how much power the device needs.

LLDP-MED also supports an extended power TLV to advertise fine-grained power requirements, end-point power priority, and end-point and network connectivity-device power status. LLDP is enabled and power is applied to a port, the power TLV determines the actual power requirement of the endpoint device so that the system power budget can be adjusted accordingly. The device processes the requests and either grants or denies power based on the current power budget. If the request is granted, the switch updates the power budget. If the request is denied, the device turns off power to the port, generates a syslog

message, and updates the power budget. If LLDP-MED is disabled or if the endpoint does not support the LLDP-MED power TLV, the initial allocation value is used throughout the duration of the connection.

You can change power settings by entering the **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} interface configuration command. By default the PoE interface is in **auto** mode; If no value is specified, the maximum is allowed (30 W).

- Inventory management TLV

Allows an endpoint to send detailed inventory information about itself to the device, including information hardware revision, firmware version, software version, serial number, manufacturer name, model name, and asset ID TLV.

- Location TLV

Provides location information from the device to the endpoint device. The location TLV can send this information:

- Civic location information

Provides the civic address information and postal information. Examples of civic location information are street address, road name, and postal community name information.

- ELIN location information

Provides the location information of a caller. The location is determined by the Emergency location identifier number (ELIN), which is a phone number that routes an emergency call to the local public safety answering point (PSAP) and which the PSAP can use to call back the emergency caller.

Wired Location Service

The device uses the location service feature to send location and attachment tracking information for its connected devices to a Cisco Mobility Services Engine (MSE). The tracked device can be a wireless endpoint, a wired endpoint, or a wired device or controller. The device notifies the MSE of device link up and link down events through the Network Mobility Services Protocol (NMSP) location and attachment notifications.

The MSE starts the NMSP connection to the device, which opens a server port. When the MSE connects to the device there are a set of message exchanges to establish version compatibility and service exchange information followed by location information synchronization. After connection, the device periodically sends location and attachment notifications to the MSE. Any link up or link down events detected during an interval are aggregated and sent at the end of the interval.

When the device determines the presence or absence of a device on a link-up or link-down event, it obtains the client-specific information such as the MAC address, IP address, and username. If the client is LLDP-MED- or CDP-capable, the device obtains the serial number and UDI through the LLDP-MED location TLV or CDP.

Depending on the device capabilities, the device obtains this client information at link up:

- Slot and port specified in port connection
- MAC address specified in the client MAC address
- IP address specified in port connection
- 802.1X username if applicable

- Device category is specified as a *wired station*
- State is specified as *new*
- Serial number, UDI
- Model number
- Time in seconds since the device detected the association

Depending on the device capabilities, the device obtains this client information at link down:

- Slot and port that was disconnected
- MAC address
- IP address
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *delete*
- Serial number, UDI
- Time in seconds since the device detected the disassociation

When the device shuts down, it sends an attachment notification with the state *delete* and the IP address before closing the NMSP connection to the MSE. The MSE interprets this notification as disassociation for all the wired clients associated with the device.

If you change a location address on the device, the device sends an NMSP location notification message that identifies the affected ports and the changed address information.

Default LLDP Configuration

Table 5: Default LLDP Configuration

Feature	Default Setting
LLDP global state	Disabled
LLDP holdtime (before discarding)	120 seconds
LLDP timer (packet update frequency)	30 seconds
LLDP reinitialization delay	2 seconds
LLDP tlv-select	Disabled to send and receive all TLVs
LLDP interface state	Disabled
LLDP receive	Disabled
LLDP transmit	Disabled

Feature	Default Setting
LLDP med-tlv-select	Disabled to send all LLDP-MED TLVs. When LLDP is glob LLDP-MED-TLV is also enabled.

Restrictions for LLDP

- If the interface is configured as a tunnel port, LLDP is automatically disabled.
- If you first configure a network-policy profile on an interface, you cannot apply the **switchport voice vlan** command on the interface. If the **switchport voice vlan *vlan-id*** is already configured on an interface, you can apply a network-policy profile on the interface. This way the interface has the voice or voice-signaling VLAN network-policy profile applied on the interface.
- You cannot configure static secure MAC addresses on an interface that has a network-policy profile.
- When Cisco Discovery Protocol and LLDP are both in use within the same switch, it is necessary to disable LLDP on interfaces where Cisco Discovery Protocol is in use for power negotiation. LLDP can be disabled at interface level with the commands **no lldp tlv-select power-management** or **no lldp transmit / no lldp receive**.

How to Configure LLDP, LLDP-MED, and Wired Location Service

Enabling LLDP

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp run**
4. **interface *interface-id***
5. **lldp transmit**
6. **lldp receive**
7. **end**
8. **show lldp**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	lldp run Example: Switch (config)# lldp run	Enables LLDP globally on the device.
Step 4	interface <i>interface-id</i> Example: Switch (config)# interface gigabitethernet 2/0/1	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 5	lldp transmit Example: Switch(config-if)# lldp transmit	Enables the interface to send LLDP packets.
Step 6	lldp receive Example: Switch(config-if)# lldp receive	Enables the interface to receive LLDP packets.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 8	show lldp Example: Switch# show lldp	Verifies the configuration.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring LLDP Characteristics

You can configure the frequency of LLDP updates, the amount of time to hold the information before discarding it, and the initialization delay time. You can also select the LLDP and LLDP-MED TLVs to send and receive.



Note Steps 3 through 6 are optional and can be performed in any order.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp holdtime *seconds***
4. **lldp reinit *delay***
5. **lldp timer *rate***
6. **lldp tlv-select**
7. **interface *interface-id***
8. **lldp med-tlv-select**
9. **end**
10. **show lldp**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	lldp holdtime <i>seconds</i> Example: Switch(config)# lldp holdtime 120	(Optional) Specifies the amount of time a receiving device should hold the information from your device before discarding it. The range is 0 to 65535 seconds; the default is 120 seconds.
Step 4	lldp reinit <i>delay</i> Example:	(Optional) Specifies the delay time in seconds for LLDP to initialize on an interface. The range is 2 to 5 seconds; the default is 2 seconds.

	Command or Action	Purpose
	<code>Switch(config)# lldp reinit 2</code>	
Step 5	<p>lldp timer rate</p> <p>Example:</p> <pre>Switch(config)# lldp timer 30</pre>	<p>(Optional) Sets the sending frequency of LLDP updates in seconds.</p> <p>The range is 5 to 65534 seconds; the default is 30 seconds.</p>
Step 6	<p>lldp tlv-select</p> <p>Example:</p> <pre>Switch(config)# tlv-select</pre>	(Optional) Specifies the LLDP TLVs to send or receive.
Step 7	<p>interface interface-id</p> <p>Example:</p> <pre>Switch (config)# interface gigabitethernet 2/0/1</pre>	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 8	<p>lldp med-tlv-select</p> <p>Example:</p> <pre>Switch (config-if)# lldp med-tlv-select inventory management</pre>	(Optional) Specifies the LLDP-MED TLVs to send or receive.
Step 9	<p>end</p> <p>Example:</p> <pre>Switch (config-if)# end</pre>	Returns to privileged EXEC mode.
Step 10	<p>show lldp</p> <p>Example:</p> <pre>Switch# show lldp</pre>	Verifies the configuration.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring LLDP-MED TLVs

By default, the device only sends LLDP packets until it receives LLDP-MED packets from the end device. It then sends LLDP packets with MED TLVs, as well. When the LLDP-MED entry has been aged out, it again only sends LLDP packets.

By using the **lldp** interface configuration command, you can configure the interface not to send the TLVs listed in the following table.

Table 6: LLDP-MED TLVs

LLDP-MED TLV	Description
inventory-management	LLDP-MED inventory management TLV
location	LLDP-MED location TLV
network-policy	LLDP-MED network policy TLV
power-management	LLDP-MED power management TLV

Follow these steps to enable a TLV on an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lldp med-tlv-select**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch (config)# interface</pre>	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.

	Command or Action	Purpose
	<code>gigabitethernet 2/0/1</code>	
Step 4	lldp med-tlv-select Example: <pre>Switch(config-if)# lldp med-tlv-select inventory management</pre>	Specifies the TLV to enable.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Network-Policy TLV

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **network-policy profile** *profile number*
4. **{voice | voice-signaling} vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**]
5. **exit**
6. **interface** *interface-id*
7. **network-policy** *profile number*
8. **lldp med-tlv-select network-policy**
9. **end**
10. **show network-policy profile**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	network-policy profile <i>profile number</i> Example: Switch(config)# network-policy profile 1	Specifies the network-policy profile number, and enter network-policy configuration mode. The range is 1 to 4294967295.
Step 4	{voice voice-signaling} vlan [<i>vlan-id</i> { <i>cos cvalue</i> dscp dvalue }] [[dot1p { <i>cos cvalue</i> dscp dvalue }] none untagged] Example: Switch(config-network-policy)# voice vlan 100 cos 4	Configures the policy attributes: <ul style="list-style-type: none"> • voice—Specifies the voice application type. • voice-signaling—Specifies the voice-signaling application type. • vlan—Specifies the native VLAN for voice traffic. • <i>vlan-id</i>—(Optional) Specifies the VLAN for voice traffic. The range is 1 to 4094. • <i>cos cvalue</i>—(Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5. • dscp dvalue—(Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46. • dot1p—(Optional) Configures the telephone to use IEEE 802.1p priority tagging and use VLAN 0 (the native VLAN). • none—(Optional) Do not instruct the IP telephone about the voice VLAN. The telephone uses the configuration from the telephone key pad. • untagged—(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone. • untagged—(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone.

	Command or Action	Purpose
Step 5	exit Example: Switch(config)# exit	Returns to global configuration mode.
Step 6	interface interface-id Example: Switch (config)# interface gigabitethernet 2/0/1	Specifies the interface on which you are configuring a network-policy profile, and enter interface configuration mode.
Step 7	network-policy profile number Example: Switch(config-if)# network-policy 1	Specifies the network-policy profile number.
Step 8	lldp med-tlv-select network-policy Example: Switch(config-if)# lldp med-tlv-select network-policy	Specifies the network-policy TLV.
Step 9	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 10	show network-policy profile Example: Switch# show network-policy profile	Verifies the configuration.
Step 11	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Location TLV and Wired Location Service

Beginning in privileged EXEC mode, follow these steps to configure location information for an endpoint and to apply it to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **location** {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string identifier id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}
3. **exit**
4. **interface** *interface-id*
5. **location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**}}
6. **end**
7. Use one of the following:
 - **show location admin-tag** *string*
 - **show location civic-location identifier** *id*
 - **show location elin-location identifier** *id*
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	location { admin-tag <i>string</i> civic-location identifier { <i>id</i> host } elin-location <i>string identifier id</i> custom-location identifier { <i>id</i> host } geo-location identifier { <i>id</i> host }} Example: <pre>Switch(config)# location civic-location identifier 1 Switch(config-civic)# number 3550 Switch(config-civic)# primary-road-name "Cisco Way" Switch(config-civic)# city "San Jose" Switch(config-civic)# state CA Switch(config-civic)# building 19 Switch(config-civic)# room C6 Switch(config-civic)# county "Santa Clara" Switch(config-civic)# country US</pre>	Specifies the location information for an endpoint. <ul style="list-style-type: none"> • admin-tag—Specifies an administrative tag or site information. • civic-location—Specifies civic location information. • elin-location—Specifies emergency location information (ELIN). • custom-location—Specifies custom location information. • geo-location—Specifies geo-spatial location information. • identifier <i>id</i>—Specifies the ID for the civic, ELIN, custom, or geo location. • host—Specifies the host civic, custom, or geo location. • <i>string</i>—Specifies the site or location information in alphanumeric format.
Step 3	exit Example:	Returns to global configuration mode.

	Command or Action	Purpose
	Switch(config-civic)# exit	
Step 4	interface <i>interface-id</i> Example: Switch (config)# interface gigabitethernet2/0/1	Specifies the interface on which you are configuring the location information, and enter interface configuration mode.
Step 5	location { additional-location-information <i>word</i> civic-location-id { <i>id</i> host } elin-location-id <i>id</i> custom-location-id { <i>id</i> host } geo-location-id { <i>id</i> host } } Example: Switch(config-if)# location elin-location-id 1	Enters location information for an interface: <ul style="list-style-type: none"> • additional-location-information—Specifies additional information for a location or place. • civic-location-id—Specifies global civic location information for an interface. • elin-location-id—Specifies emergency location information for an interface. • custom-location-id—Specifies custom location information for an interface. • geo-location-id—Specifies geo-spatial location information for an interface. • host—Specifies the host location identifier. • <i>word</i>—Specifies a word or phrase with additional location information. • <i>id</i>—Specifies the ID for the civic, ELIN, custom, or geo location. The ID range is 1 to 4095.
Step 6	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 7	Use one of the following: <ul style="list-style-type: none"> • show location admin-tag <i>string</i> • show location civic-location identifier <i>id</i> • show location elin-location identifier <i>id</i> Example: Switch# show location admin-tag or Switch# show location civic-location	Verifies the configuration.

	Command or Action	Purpose
	<pre>identifier</pre> <p>OR</p> <pre>Switch# show location elin-location identifier</pre>	
Step 8	<pre>copy running-config startup-config</pre> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling Wired Location Service on the Device

Before you begin

For wired location to function, you must first enter the **ip device tracking** global configuration command.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `nmosp notification interval {attachment | location} interval-seconds`
4. `end`
5. `show network-policy profile`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example:</p> <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<pre>configure terminal</pre> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<pre>nmosp notification interval {attachment location} interval-seconds</pre> <p>Example:</p>	Specifies the NMSP notification interval. attachment —Specifies the attachment notification interval.

	Command or Action	Purpose
	Switch(config)# nmsp notification interval location 10	location —Specifies the location notification interval. <i>interval-seconds</i> —Duration in seconds before the device sends the MSE the location or attachment updates. The range is 1 to 30; the default is 30.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show network-policy profile Example: Switch# show network-policy profile	Verifies the configuration.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuration Examples for LLDP, LLDP-MED, and Wired Location Service

Configuring Network-Policy TLV: Examples

This example shows how to configure VLAN 100 for voice application with CoS and to enable the network-policy profile and network-policy TLV on an interface:

```
Switch# configure terminal
Switch(config)# network-policy 1
Switch(config-network-policy)# voice vlan 100 cos 4
Switch(config-network-policy)# exit
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# network-policy profile 1
Switch(config-if)# lldp med-tlv-select network-policy
```

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
Switchconfig-network-policy)# voice vlan dot1p cos 4
Switchconfig-network-policy)# voice vlan dot1p dscp 34
```

Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service

Commands for monitoring and maintaining LLDP, LLDP-MED, and wired location service.

Command	Description
clear lldp counters	Resets the traffic counters to zero.
clear lldp table	Deletes the LLDP neighbor information table.
clear nmsp statistics	Clears the Nmsp statistic counters.
show lldp	Displays global information, such as frequency of transmissions, the holdtime for packets being sent, and the delay time before LLDP initializes on an interface.
show lldp entry <i>entry-name</i>	Displays information about a specific neighbor. You can enter an asterisk (*) to display all neighbors, or you can enter the neighbor name.
show lldp interface [<i>interface-id</i>]	Displays information about interfaces with LLDP enabled. You can limit the display to a specific interface.
show lldp neighbors [<i>interface-id</i>] [detail]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, and port ID. You can limit the display to neighbors of a specific interface or expand the display for more detailed information.
show lldp traffic	Displays LLDP counters, including the number of packets sent and received, number of packets discarded, and number of unrecognized TLVs.
show location admin-tag <i>string</i>	Displays the location information for the specified administrative tag or site.
show location civic-location identifier <i>id</i>	Displays the location information for a specific global civic location.
show location elin-location identifier <i>id</i>	Displays the location information for an emergency location
show network-policy profile	Displays the configured network-policy profiles.
show nmsp	Displays the Nmsp information



CHAPTER 4

Configuring MultiGigabit Ports on WS-C3560CX-8PD-S

- [Finding Feature Information, on page 47](#)
- [Overview of MultiGigabit Ports, on page 47](#)
- [Restrictions for MultiGigabit Ports, on page 48](#)
- [Supported Cable Types and Maximum Length, on page 48](#)
- [Setting the Interface Speed, on page 48](#)
- [Examples: Setting the Interface Speed, on page 49](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Overview of MultiGigabit Ports

Cisco's MultiGigabit Ethernet technology allows you to leverage 802.11ac Wave 2 speeds on your device. Beginning with Cisco IOS XE 3.7.E1 and IOS 15.2(3) E1, you can configure the WS-C3560CX-8XPD-S module to auto-negotiate multiple speeds on switch ports, and support 100 Mbps, 1 Gbps, 2.5 Gbps, and 5 Gbps speeds on Category 5e cables, and up to 10 Gbps over Category6 and Category 6a cables.

Beginning with Cisco IOS XE 3.9.E1 and IOS 15.2(5) E1, by default, interface speeds are downshifted on multiGigabit ports. When an interface is unable to establish a high speed link, the line rate is automatically downshifted or reduced to a lower speed. The interface tries up to four times to reestablish a link using the current speed, before downshifting to the next available lower speed. For multiGigabit interfaces to support downshifting, the interface speed must be set to **auto** on both sides of the link.

The WS-C3560CX-8XPD-S module has 8 ports, of which the 6 ports are 1-Gigabit Ethernet ports and 2 ports are multiGigabit ports. The module also has 2 SFP+ ports.

Restrictions for MultiGigabit Ports

The following restrictions apply:

- MultiGigabit ports do not support 10Mbps speed.
- MultiGigabit ports do not support half-duplex mode.
- MultiGigabit ports do not support EEE.
- Multigigabit ports support downshift only when interface speed is set to **auto** on both sides of the link.

Supported Cable Types and Maximum Length

The following table lists the types of cables and the maximum length of cables supported on the Multigigabit ports.

Cable Type	100M	1G	2.5G	5G	10G
Category5E	Yes	Yes	Yes	Yes	Not Available
Category6	Yes	Yes	Yes	Yes	Yes (55 meters)
Category6A	Yes	Yes	Yes	Yes	Yes

Setting the Interface Speed

To set port speed to 100Mbps/1000Mbps/2500Mbps/5000Mbps/10000Mbps on a Multigigabit Ethernet interface (on a 1000Base-T port), perform this task:



Note Only 2 ports on the WS-C3560CX-8XPD-S module support Multigigabit Ethernet.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tengigabitethernet slot/interface**
4. **speed [100 | 1000 | 2500 | 5000 | 10000 | auto [100 | 1000 | 2500 | 5000 | 10000]]**
5. **[no] downshift disable**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: <pre>Switch> enable</pre>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface tengigabitethernet slot/interface Example: <pre>Switch(config)# interface tengigabitethernet 1/0/2</pre>	Specifies the interface to be configured.
Step 4	speed [100 1000 2500 5000 10000 auto [100 1000 2500 5000 10000]] Example: <pre>Switch (config-if)# speed 5000</pre>	Sets the interface speed. Note 10G speed is supported only on Category6 and Category6a cables.
Step 5	[no] downshift disable Example: <pre>Switch(config-if)# no downshift disable</pre>	By default, downshift is enabled on multigigabit ports. The downshift disable command disables downshift on the specified interface. The no downshift disable command enables downshift on the interface.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

What to do next

To restore autonegotiation (default setting), enter the **no speed** command in the interface configuration mode.

Examples: Setting the Interface Speed

This example shows how to set the interface speed to 5G on the Multigigabit Ethernet interface 1/0/2:

```
Switch(config)# interface tengigabitethernet 1/0/2
Switch (config-if)# speed 5000
```

This example shows how to allow the Multigigabit Ethernet interface 1/0/2 to autonegotiate the speed and duplex mode:

```
Switch(config)# interface tengigabitethernet 1/0/2
Switch(config-if)# speed auto
```

This example shows how to limit speed negotiation to 2.5G on the Multigigabit Ethernet interface 1/0/1:

```
Switch(config)# interface tengigabitethernet 1/0/1
Switch(config-if)# speed auto 2500
```



CHAPTER 5

Configuring System MTU

- [Information About the MTU, on page 51](#)
- [How to Configure MTU , on page 51](#)
- [Configuration Examples for System MTU, on page 52](#)

Information About the MTU

System MTU Guidelines

When configuring the system MTU values, follow these guidelines:

- The default maximum transmission unit (MTU) size for frames received and transmitted on all interfaces is 1500 bytes. You can increase the MTU size for all interfaces operating at 10 or 100 Mb/s by using the **system mtu** global configuration command. You can increase the MTU size to support jumbo frames on all Gigabit Ethernet interfaces by using the **system mtu jumbo** global configuration command.
- Gigabit Ethernet ports are not affected by the **system mtu** command; 10/100 ports are not affected by the **system mtu jumbo** command. If you do not configure the **system mtu jumbo** command, the setting of the **system mtu** command applies to all Gigabit Ethernet interfaces.

How to Configure MTU

Configuring the System MTU

Beginning in privileged EXEC mode, follow these steps to change the MTU size for all 10/100 or Gigabit Ethernet interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **system mtu *bytes***
3. **system mtu jumbo *bytes***
4. **end**

5. `copy running-config startup-config`
6. `show system mtu`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	system mtu bytes Example: Switch(config)# <code>system mtu 2500</code>	The range is 1500 to 9198 bytes; the default is 1500 bytes.
Step 3	system mtu jumbo bytes Example: Switch(config)# <code>system mtu jumbo7500</code>	The range is 1500 to 9198 bytes; the default is 1500 bytes.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	Saves your entries in the configuration file.
Step 6	show system mtu Example: Switch# <code>show system mtu</code>	Verifies your settings.

Configuration Examples for System MTU

This example shows how to set the maximum packet size for a Gigabit Ethernet port to bytes:

```
Switch(config)#
Switch(config)# exit
```



CHAPTER 6

Configuring Boot Fast

- [Configuring Boot Fast on the switch, on page 53](#)

Configuring Boot Fast on the switch

This feature, when enabled, helps the switch to boot up fast. The Memory test is performed for a limited range, the switch skips File system check (FSCK) and Skips Post test.



Note When Fast boot is enabled, you can still run the POST tests manually from the command line interface, once the switch has booted up, using **diagnostic start** command.

Enabling Boot Fast

To enable the boot fast feature, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **boot fast**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	boot fast Example: Switch(config)# <code>boot fast</code>	Enables fast boot feature Performs Memory test for a limited range, Skips File system check (FSCK) and Skips Post test.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.

Disabling Boot Fast

To disable the boot fast feature, perform the following steps:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `no boot fast`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no boot fast Example: Switch(config)# <code>no boot fast</code>	Disables the boot fast feature.

	Command or Action	Purpose
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.



CHAPTER 7

Configuring Power over Ethernet

- [Information About PoE, on page 57](#)
- [How to Configure PoE, on page 63](#)
- [Monitoring Power Status, on page 72](#)
- [Configuration Examples for Configuring PoE, on page 72](#)

Information About PoE

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the device senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

PoE and PoE Pass-Through Ports on Catalyst WS-C3560CX-8PT-S

The Catalyst WS-C3560CX-8PT-S is a PD/PSE product, which means that the switch can behave like both a Power Device (PD) and Power Source Equipment (PSE). This switch will be powered on by the PoE voltage derived from its uplink ports (PD1 or PD2) or from the voltage supplied by external auxiliary power supply (AUX). The switch will enable powering over PoE, PoE+ and UPOE, as well as AC and DC input.

The power available from uplinks and one of the power adapter will be added for increased input power, which translates to a higher PoE budget. Some of this power will be used for system power and rest would be provided to downlink POE+ ports as pass-through power that will be available to power other PoE peripheral devices like IP phones, IP Cameras and so on.

- The Catalyst WS-C3560CX-8PT-S will support powering from 2xUPOE uplinks.
- It will support a DC power adaptor which will enable the switch to be powered by 24V DC input.
- AUX contributes 78W to the system.

- The power sources (AC or DC) and PoE will be additive. The table below lists different power values for PoE budget.

Table 7: PoE Budget

PoE Budget(Watts)	Uplink 1	Uplink 2	Comment
0	PoE	PoE	Normal operation, no PoE budget
0	0	PoE+	Normal operation, no PoE budget
20	PoE+	PoE+	PoE budget available
22	0	UPoE	PoE budget available
33	UPoE	PoE	PoE budget available
44	PoE+	UPoE	PoE budget available
68	UPoE	UPoE	PoE budget available

The switch is expected to boot with T1 power and negotiate to T2 power which is known as Low Power Bootup. The Low Power Bootup occurs in the following case:

- One of the uplink port is connected to the PSE.
- No Auxiliary power adapter is connected.

In this case, the switch will power up in low power mode with ASIC powered down and negotiate power using CDP/LLDP. The system will power up and initialize ASIC once power is negotiated and continue to boot without software reload.

Example: Configuring PoE and PoE Pass-Through Ports on WS-C3560CX-8PT-S

The **show env power** privileged EXEC command provides information about powering options on your switch:

```
Switch# show env power
```

```
Power Source  Type          Power(w)  Mode
-----
A.C. Input    Auxilliary    80(w)     Available
Gi0/9         Type2         30(w)     Available
Gi0/10        Type2         30(w)     Available
```

Available : The PoE received on this link is used for powering this switch and providing PoE pass-through if applicable.



Note All these power sources adds up to the POE budget. The system consumption is approximately 24W.

Supported Protocols and Standards

The device uses these protocols and standards to support PoE:

- CDP with power consumption—The powered device notifies the device of the amount of power it is consuming. The device does not reply to the power-consumption messages. The device can only supply power to or remove power from the PoE port.
- Cisco intelligent power management—The powered device and the device negotiate through power-negotiation CDP messages for an agreed-upon power-consumption level. The negotiation allows a high-power Cisco powered device, which consumes more than 7 W, to operate at its highest power mode. The powered device first boots up in low-power mode, consumes less than 7 W, and negotiates to obtain enough power to operate in high-power mode. The device changes to high-power mode only when it receives confirmation from the device.

High-power devices can operate in low-power mode on devices that do not support power-negotiation CDP.

Cisco intelligent power management is backward-compatible with CDP with power consumption; the device responds according to the CDP message that it receives. CDP is not supported on third-party powered devices; therefore, the device uses the IEEE classification to determine the power usage of the device.

- IEEE 802.3af—The major features of this standard are powered-device discovery, power administration, disconnect detection, and optional powered-device power classification. For more information, see the standard.

Powered-Device Detection and Initial Power Allocation

The device detects a Cisco pre-standard or an IEEE-compliant powered device when the PoE-capable port is in the no-shutdown state, PoE is enabled (the default), and the connected device is not being powered by an AC adaptor.

After device detection, the device determines the device power requirements based on its type:

- The initial power allocation is the maximum amount of power that a powered device requires. The device initially allocates this amount of power when it detects and powers the powered device. As the device receives CDP messages from the powered device and as the powered device negotiates power levels with the device through CDP power-negotiation messages, the initial power allocation might be adjusted.
- The device classifies the detected IEEE device within a power consumption class. Based on the available power in the power budget, the device determines if a port can be powered. [Table 8: IEEE Power Classifications, on page 59](#) lists these levels.

Table 8: IEEE Power Classifications

Class	Maximum Power Level Required from the Device
0 (class status unknown)	15.4 W
1	4 W
2	7 W
3	15.4 W

The device monitors and tracks requests for power and grants power only when it is available. The device tracks its power budget (the amount of power available on the device for PoE). The device performs power-accounting calculations when a port is granted or denied power to keep the power budget up to date.

After power is applied to the port, the device uses CDP to determine the *CDP-specific* power consumption requirement of the connected Cisco powered devices, which is the amount of power to allocate based on the CDP messages. The device adjusts the power budget accordingly. This does not apply to third-party PoE devices. The device processes a request and either grants or denies power. If the request is granted, the device updates the power budget. If the request is denied, the device ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. Powered devices can also negotiate with the device for more power.

If the device detects a fault caused by an undervoltage, overvoltage, overtemperature, oscillator-fault, or short-circuit condition, it turns off power to the port, generates a syslog message, and updates the power budget and LEDs.

Power Management Modes

The device supports these PoE modes:

- **auto**—The device automatically detects if the connected device requires power. If the device discovers a powered device connected to the port and if the device has enough power, it grants power, updates the power budget, turns on power to the port on a first-come, first-served basis, and updates the LEDs. For LED information, see the hardware installation guide.

If the device has enough power for all the powered devices, they all come up. If enough power is available for all powered devices connected to the device, power is turned on to all devices. If there is not enough available PoE, or if a device is disconnected and reconnected while other devices are waiting for power, it cannot be determined which devices are granted or are denied power.

If granting power would exceed the system power budget, the device denies power, ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. After power has been denied, the device periodically rechecks the power budget and continues to attempt to grant the request for power.

If a device being powered by the device is then connected to wall power, the device might continue to power the device. The device might continue to report that it is still powering the device whether the device is being powered by the device or receiving power from an AC power source.

If a powered device is removed, the device automatically detects the disconnect and removes power from the port. You can connect a nonpowered device without damaging it.

You can specify the maximum wattage that is allowed on the port. If the IEEE class maximum wattage of the powered device is greater than the configured maximum value, the device does not provide power to the port. If the device powers a powered device, but the powered device later requests through CDP messages more than the configured maximum value, the device removes power to the port. The power that was allocated to the powered device is reclaimed into the global power budget. If you do not specify a wattage, the device delivers the maximum value. Use the **auto** setting on any PoE port. The auto mode is the default setting.

- **static**—The device pre-allocates power to the port (even when no powered device is connected) and guarantees that power will be available for the port. The device allocates the port configured maximum wattage, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed to be powered when it is connected to the static port. The port no longer participates in the first-come, first-served model.

However, if the powered-device IEEE class is greater than the maximum wattage, the device does not supply power to it. If the device learns through CDP messages that the powered device is consuming more than the maximum wattage, the device shuts down the powered device.

If you do not specify a wattage, the device pre-allocates the maximum value. The device powers the port only if it discovers a powered device. Use the **static** setting on a high-priority interface.

- **never**—The device disables powered-device detection and never powers the PoE port even if an unpowered device is connected. Use this mode only when you want to make sure that power is never applied to a PoE-capable port, making the port a data-only port.

For most situations, the default configuration (auto mode) works well, providing plug-and-play operation. No further configuration is required. However, perform this task to configure a PoE port for a higher priority, to make it data only, or to specify a maximum wattage to disallow high-power powered devices on a port.

Power Monitoring and Power Policing

When policing of the real-time power consumption is enabled, the device takes action when a powered device consumes more power than the maximum amount allocated, also referred to as the *cutoff-power value*.

When PoE is enabled, the device senses the real-time power consumption of the powered device. The device monitors the real-time power consumption of the connected powered device; this is called *power monitoring* or *power sensing*. The device also polices the power usage with the *power policing* feature.

Power monitoring is backward-compatible with Cisco intelligent power management and CDP-based power consumption. It works with these features to ensure that the PoE port can supply power to the powered device.

The device senses the real-time power consumption of the connected device as follows:

1. The device monitors the real-time power consumption on individual ports.
2. The device records the power consumption, including peak power usage. The device reports the information through the CISCO-POWER-ETHERNET-EXT-MIB.
3. If power policing is enabled, the device polices power usage by comparing the real-time power consumption to the maximum power allocated to the device. The maximum power consumption is also referred to as the *cutoff power* on a PoE port.

If the device uses more than the maximum power allocation on the port, the device can either turn off power to the port, or the device can generate a syslog message and update the LEDs (the port LED is now blinking amber) while still providing power to the device based on the device configuration. By default, power-usage policing is disabled on all PoE ports.

If error recovery from the PoE error-disabled state is enabled, the device automatically takes the PoE port out of the error-disabled state after the specified amount of time.

If error recovery is disabled, you can manually re-enable the PoE port by using the **shutdown** and **no shutdown** interface configuration commands.

4. If policing is disabled, no action occurs when the powered device consumes more than the maximum power allocation on the PoE port, which could adversely affect the device.

Maximum Power Allocation (Cutoff Power) on a PoE Port

When power policing is enabled, the device determines one of the these values as the cutoff power on the PoE port in this order:

1. Manually when you set the user-defined power level that the device budgets for the port by using the **power inline consumption default** *wattage* global or interface configuration command
2. Manually when you set the user-defined power level that limits the power allowed on the port by using the **power inline auto max** *max-wattage* or the **power inline static max** *max-wattage* interface configuration command
3. Automatically when the device sets the power usage of the device by using CDP power negotiation or by the IEEE classification and LLDP power negotiation.

Use the first or second method in the previous list to manually configure the cutoff-power value by entering the **power inline consumption default** *wattage* or the **power inline** [**auto** | **static max**] *max-wattage* command.

You should use **power inline consumption default** *wattage* command to manually set the power level for a port only in situations where CDP/LLDP power negotiations are not supported.

If you do not manually configure the cutoff-power value, the device automatically determines it by using CDP power negotiation or the device IEEE classification and LLDP power negotiation. If CDP or LLDP are not enabled, the default value of 30 W is applied. However without CDP or LLDP, the device does not allow devices to consume more than 15.4 W of power because values from 15400 to 30000 mW are only allocated based on CDP or LLDP requests. If a powered device consumes more than 15.4 W without CDP or LLDP negotiation, the device might be in violation of the maximum current (*I_{max}*) limitation and might experience an *I_{cut}* fault for drawing more current than the maximum. The port remains in the fault state for a time before attempting to power on again. If the port continuously draws more than 15.4 W, the cycle repeats.



Note When a powered device connected to a PoE+ port restarts and sends a CDP or LLDP packet with a power TLV, the device locks to the power-negotiation protocol of that first packet and does not respond to power requests from the other protocol. For example, if the device is locked to CDP, it does not provide power to devices that send LLDP requests. If CDP is disabled after the device has locked on it, the device does not respond to LLDP power requests and can no longer power on any accessories. In this case, you should restart the powered device.

Power Consumption Values

You can configure the initial power allocation and the maximum power allocation on a port. However, these values are only the configured values that determine when the device should turn on or turn off power on the PoE port. The maximum power allocation is not the same as the actual power consumption of the powered device. The actual cutoff power value that the device uses for power policing is not equal to the configured power value.

When power policing is enabled, the device polices the power usage *at the switch port*, which is greater than the power consumption of the device. When you manually set the maximum power allocation, you must consider the power loss over the cable from the switch port to the powered device. The cutoff power is the sum of the rated power consumption of the powered device and the worst-case power loss over the cable.

We recommend that you enable power policing when PoE is enabled on your device. For example, if policing is disabled and you set the cutoff-power value by using the **power inline auto max 6300** interface configuration command, the configured maximum power allocation on the PoE port is 6.3 W (6300 mW). The device provides power to the connected devices on the port if the device needs up to 6.3 W. If the CDP-power negotiated value or the IEEE classification value exceeds the configured cutoff value, the device does not provide power to the connected device. After the device turns on power on the PoE port, the device does not police the real-time power consumption of the device, and the device can consume more power than the

maximum allocated amount, which could adversely affect the device and the devices connected to the other PoE ports.

Because the device supports internal power supplies and the Cisco Redundant Power System 2300 (also referred to as the RPS 2300), the total amount of power available for the powered devices varies depending on the power supply configuration.

How to Configure PoE

Configuring a Power Management Mode on a PoE Port



Note When you make PoE configuration changes, the port being configured drops power. Depending on the new configuration, the state of the other PoE ports, and the state of the power budget, the port might not be powered up again. For example, port 1 is in the auto and on state, and you configure it for static mode. The device removes power from port 1, detects the powered device, and repowers the port. If port 1 is in the auto and on state and you configure it with a maximum wattage of 10 W, the device removes power from the port and then redetects the powered device. The device repowers the port only if the powered device is a class 1, class 2, or a Cisco-only powered device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]}
5. **end**
6. **show power inline** [*interface-id* | **module** *switch-number*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example:	Specifies the physical port to be configured, and enters interface configuration mode.

	Command or Action	Purpose
	Switch(config)# interface gigabitethernet 2/0/1	
Step 4	<p>power inline {auto [max <i>max-wattage</i>] never static [max <i>max-wattage</i>]}</p> <p>Example:</p> <pre>Switch(config-if)# power inline auto</pre>	<p>Configures the PoE mode on the port. The keywords have these meanings:</p> <ul style="list-style-type: none"> • auto—Enables powered-device detection. If enough power is available, automatically allocates power to the PoE port after device detection. This is the default setting. • max <i>max-wattage</i>—Limits the power allowed on the port. If no value is specified, the maximum is allowed. • never—Disables device detection, and disable power to the port. <p>Note If a port has a Cisco powered device connected to it, do not use the power inline never command to configure the port. A false link-up can occur, placing the port into the error-disabled state.</p> <ul style="list-style-type: none"> • static—Enables powered-device detection. Pre-allocate (reserve) power for a port before the device discovers the powered device. The device reserves power for this port even when no device is connected and guarantees that power will be provided upon device detection. <p>The device allocates power to a port configured in static mode before it allocates power to a port configured in auto mode.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show power inline [<i>interface-id</i> module <i>switch-number</i>]</p> <p>Example:</p> <pre>Switch# show power inline</pre>	Displays PoE status for a device, for the specified interface.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring PoE and PoE Pass-Through Ports on Catalyst WS-C3560CX-8PT-S

You can configure the power management, budgeting, and policing on the Catalyst WS-C3560CX-8PT-S compact switch PoE ports the same as with any other PoE switch.

The **show env power** privileged EXEC command provides information about powering options on your switch.

Perpetual POE

Perpetual POE provides uninterrupted power to connected PD device even when the PSE switch is booting.



Note Power to the ports will be interrupted in case of MCU firmware upgrade and ports will be back up immediately after the upgrade.



Note This feature is available only on the following models of Catalyst 3560-CX and Catalyst 2960-CX switches:

- WS-3560CX-8PC-S
- WS-3560CX-12PC-S
- WS-C3560CX-8XPD-S
- WS-C2960CX-8PC-L

Fast POE

This feature remembers the last power drawn from a particular PSE port and switches on power the moment AC power is plugged in (within 15 to 20 seconds of switching on power) without waiting for IOS to boot up. When **poe-ha** is enabled on a particular port, the switch on a recovery after power failure, provides power to the connected endpoint devices within short duration before even the IOS forwarding starts up.

This feature can be configured by the same command as **poe-ha** which is already implemented. If the user replaces the power device connected to a port when the switch is powered off, then this new device will get the power which the previous device was drawing.



Note Fast POE is supported on Catalyst 3850 only.



Note In case of UPOE, even though Fast POE is available on the switch side, the PD endpoints may not be able to take advantage of the same, due to the reliance on LLDP to signal the UPOE power availability. This reliance on LLDP requires that the PD endpoint still needs to wait till the IOS comes up and LLDP packet exchanges can happen, signaling the availability of UPOE power.

Configuring Persistent and Fast PoE

To configure persistent POE and PoE, perform the following steps:



Note You will need to configure the **poe-ha** command before connecting the PD, or you will need to manually shut/unshut the port after configuring **poe-ha**.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline port poe-ha**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/1	Specifies the physical port to be configured, and enters interface configuration mode.
Step 4	power inline port poe-ha Example: Switch(config-if)# power inline port poe-ha	Configures POE High Availability.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Budgeting Power for Devices Connected to a PoE Port

When Cisco powered devices are connected to PoE ports, the device uses Cisco Discovery Protocol (CDP) to determine the *protocol-specific* power consumption of the devices, and the device adjusts the power budget accordingly. This does not apply to IEEE third-party powered devices. For these devices, when the device grants a power request, the device adjusts the power budget according to the powered-device IEEE classification. If the powered device is a class 0 (class status unknown) or a class 3, the device budgets 15,400 mW for the device, regardless of the CDP-specific amount of power needed. If the powered device reports a higher class than its CDP-specific consumption or does not support power classification (defaults to class 0), the device can power fewer devices because it uses the IEEE class information to track the global power budget.

By using the **power inline consumption wattage** interface configuration command or the **power inline consumption default wattage** global configuration command, you can override the default power requirement specified by the IEEE classification. The difference between what is mandated by the IEEE classification and what is actually needed by the device is reclaimed into the global power budget for use by additional devices. You can then extend the device power budget and use it more effectively.



Caution You should carefully plan your device power budget, enable the power monitoring feature, and make certain not to oversubscribe the power supply.



Note When you manually configure the power budget, you must also consider the power loss over the cable between the device and the powered device.

Budgeting Power to All PoE ports

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **power inline consumption default wattage**
5. **end**
6. **show power inline consumption default**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no cdp run Example: Switch(config)# <code>no cdp run</code>	(Optional) Disables CDP.
Step 4	power inline consumption default <i>wattage</i> Example: Switch(config)# <code>power inline consumption default 5000</code>	Configures the power consumption of powered devices connected to each PoE port.
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show power inline consumption default Example: Switch# <code>show power inline consumption default</code>	Displays the power consumption status.
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Budgeting Power to a Specific PoE Port

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `no cdp run`
4. `interface interface-id`
5. `power inline consumption wattage`
6. `end`
7. `show power inline consumption`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> <code>enable</code>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no cdp run Example: Switch(config)# <code>no cdp run</code>	(Optional) Disables CDP.
Step 4	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the physical port to be configured, and enter interface configuration mode.
Step 5	power inline consumption <i>wattage</i> Example: Switch(config-if)# <code>power inline consumption 5000</code>	Configures the power consumption of a powered device connected to a PoE port on the device. The range for each device is 4000 to . The default is .
Step 6	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 7	show power inline consumption Example: Switch# <code>show power inline consumption</code>	Displays the power consumption data.
Step 8	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Power Policing

By default, the device monitors the real-time power consumption of connected powered devices. You can configure the device to police the power usage. By default, policing is disabled.

SUMMARY STEPS

1. `enable`

2. **configure terminal**
3. **interface** *interface-id*
4. **power inline police** [action {log | errdisable}]
5. **exit**
6. Use one of the following:
 - **errdisable detect cause inline-power**
 - **errdisable recovery cause inline-power**
 - **errdisable recovery interval** *interval*
7. **exit**
8. Use one of the following:
 - **show power inline police**
 - **show errdisable recovery**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 2/0/1	Specifies the physical port to be configured, and enter interface configuration mode.
Step 4	power inline police [action {log errdisable}] Example: Switch(config-if)# power inline police	If the real-time power consumption exceeds the maximum power allocation on the port, configures the device to take one of these actions: <ul style="list-style-type: none"> • power inline police—Shuts down the PoE port, turns off power to it, and puts it in the error-disabled state. <p>Note You can enable error detection for the PoE error-disabled cause by using the errdisable detect cause inline-power global configuration command. You can also enable the timer to recover from the PoE error-disabled state by using the errdisable recovery cause inline-power interval <i>interval</i> global configuration command.</p>

	Command or Action	Purpose
		<ul style="list-style-type: none"> • power inline police action errdisable—Turns off power to the port if the real-time power consumption exceeds the maximum power allocation on the port. • power inline police action log—Generates a syslog message while still providing power to the port. <p>If you do not enter the action log keywords, the default action shuts down the port and puts the port in the error-disabled state.</p>
Step 5	exit Example: <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	Use one of the following: <ul style="list-style-type: none"> • errdisable detect cause inline-power • errdisable recovery cause inline-power • errdisable recovery interval interval Example: <pre>Switch(config)# errdisable detect cause inline-power</pre> <pre>Switch(config)# errdisable recovery cause inline-power</pre> <pre>Switch(config)# errdisable recovery interval 100</pre>	(Optional) Enables error recovery from the PoE error-disabled state, and configures the PoE recover mechanism variables. By default, the recovery interval is 300 seconds. For interval interval , specifies the time in seconds to recover from the error-disabled state. The range is 30 to 86400.
Step 7	exit Example: <pre>Switch(config)# exit</pre>	Returns to privileged EXEC mode.
Step 8	Use one of the following: <ul style="list-style-type: none"> • show power inline police • show errdisable recovery Example: <pre>Switch# show power inline police</pre> <pre>Switch# show errdisable recovery</pre>	Displays the power monitoring status, and verify the error recovery settings.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring Power Status

Table 9: Show Commands for Power Status

Command	Purpose
show env power switch	(Optional) Displays the status of the internal power supplies for the specified switch.
show power inline [<i>interface-id</i>]	Displays PoE status for a switch, for an interface.
show power inline police	Displays the power policing data.
show env power	Displays the status of the power supplies for the specified switch.

Configuration Examples for Configuring PoE

Budgeting Power: Example

When you enter one of the following commands,

- **[no] power inline consumption default** *wattage* global configuration command
- **[no] power inline consumption** *wattage*
interface configuration command

this caution message appears:

```
%CAUTION: Interface Gi1/0/1: Misconfiguring the 'power inline consumption/allocation'
command may cause damage to the
switch and void your warranty. Take precaution not to oversubscribe the power supply. It
is recommended to enable power
policing if the switch supports it. Refer to documentation.
```




CHAPTER 8

Configuring 2-event Classification

- [Information about 2-event Classification, on page 73](#)
- [Configuring 2-event Classification, on page 73](#)
- [Example: Configuring 2-Event Classification, on page 74](#)

Information about 2-event Classification

When a class 4 device gets detected, IOS allocates 30W without any CDP or LLDP negotiation. This means that even before the link comes up the class 4 power device gets 30W.

Also, on the hardware level the PSE does a 2-event classification which allows a class 4 PD to detect PSE capability of providing 30W from hardware, register itself and it can move up to PoE+ level without waiting for any CDP/LLDP packet exchange.

Once 2-event is enabled on a port, you need to manually shut/un-shut the port or connect the PD again to start the IEEE detection again. Power budget allocation for a class-4 device will be 30W if 2-event classification is enabled on the port, else it will be 15.4W.

Configuring 2-event Classification

To configure the switch for a 2-event Classification, perform the steps given below:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline port 2-event**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/1	Specifies the physical port to be configured, and enters interface configuration mode.
Step 4	power inline port 2-event Example: Switch(config-if)# power inline port 2-event	Configures 2-event classification on the switch.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Example: Configuring 2-Event Classification

This example shows how you can configure 2-event classification.

```
Switch> enable
Switch# configure terminal
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# power inline port 2-event
Switch(config-if)# end
```



CHAPTER 9

Configuring EEE

- [Restrictions for EEE, on page 75](#)
- [Information About EEE, on page 75](#)
- [How to Configure EEE, on page 75](#)
- [Monitoring EEE, on page 77](#)
- [Configuration Examples for Configuring EEE, on page 77](#)

Restrictions for EEE

Energy Efficient Ethernet (EEE) has the following restrictions:

- Changing the EEE configuration resets the interface because the device has to restart Layer 1 autonegotiation.
- You might want to enable the Link Layer Discovery Protocol (LLDP) for devices that require longer wakeup times before they are able to accept data on their receive paths. Doing so enables the device to negotiate for extended system wakeup times from the transmitting link partner.

Information About EEE

EEE Overview

Energy Efficient Ethernet (EEE) is an IEEE 802.3az standard that is designed to reduce power consumption in Ethernet networks during idle periods.

Default EEE Configuration

How to Configure EEE

You can enable or disable EEE on an interface that is connected to an EEE-capable link partner.

Enabling or Disabling EEE

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **power efficient-ethernet auto**
4. **no power efficient-ethernet auto**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	power efficient-ethernet auto Example: Device(config-if)# power efficient-ethernet auto	Enables EEE on the specified interface. When EEE is enabled, the device advertises and autonegotiates EEE to its link partner.
Step 4	no power efficient-ethernet auto Example: Device(config-if)# no power efficient-ethernet auto	Disables EEE on the specified interface.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Device# <code>copy running-config startup-config</code>	

Monitoring EEE

Table 10: Commands for Displaying EEE Settings

Command	Purpose
<code>show eee capabilities interface interface-id</code>	Displays EEE capabilities for the specified interface.
<code>show eee status interface interface-id</code>	Displays EEE status information for the specified interface.
<code>show eee counters interface interface-id</code>	Displays EEE counters for the specified interface.

Following are examples of the `show eee` commands

```
Switch#show eee capabilities interface gigabitEthernet2/0/1
Gi2/0/1
EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
Link Partner : yes (100-Tx and 1000T auto)

ASIC/Interface : EEE Capable/EEE Enabled

Switch#show eee status interface gigabitEthernet2/0/1
Gi2/0/1 is up
EEE(efficient-ethernet): Operational
Rx LPI Status : Low Power
Tx LPI Status : Low Power
Wake Error Count : 0

ASIC EEE STATUS
Rx LPI Status : Receiving LPI
Tx LPI Status : Transmitting LPI
Link Fault Status : Link Up
Sync Status : Code group synchronization with data stream intact

Switch#show eee counters interface gigabitEthernet2/0/1

LP Active Tx Time (10us) : 66649648
LP Transitioning Tx : 462
LP Active Rx Time (10us) : 64911682
LP Transitioning Rx : 153
```

Configuration Examples for Configuring EEE

This example shows how to enable EEE for an interface:

```
Device# configure terminal
Device(config)# interface gigabitEthernet 1/0/1
Device(config-if)# power efficient-ethernet auto
```

This example shows how to disable EEE for an interface:

```
Device# configure terminal  
Device(config)# interface gigabitethernet 1/0/1  
Device(config-if)# no power efficient-ethernet auto
```



PART II

IP Multicast Routing

- [IP Multicast Routing Technology Overview, on page 81](#)
- [Configuring Basic IP Multicast Routing, on page 87](#)
- [Configuring IGMP, on page 99](#)
- [Configuring IGMP Snooping and Multicast VLAN Registration, on page 117](#)
- [Configuring CGMP, on page 163](#)
- [Configuring Protocol Independent Multicast \(PIM\), on page 169](#)
- [Configuring HSRP Aware PIM, on page 221](#)
- [Configuring VRRP Aware PIM, on page 227](#)
- [Configuring SSM, on page 231](#)
- [Configuring MSDP, on page 253](#)



CHAPTER 10

IP Multicast Routing Technology Overview

- [Information About IP Multicast Technology, on page 81](#)

Information About IP Multicast Technology

Role of IP Multicast in Information Delivery

IP multicast is a bandwidth-conserving technology that reduces traffic by delivering a single stream of information simultaneously to potentially thousands of businesses and homes. Applications that take advantage of multicast include video conferencing, corporate communications, distance learning, and distribution of software, stock quotes, and news.

IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address. The sending host inserts the multicast group address into the IP destination address field of the packet and IP multicast routers and multilayer switches forward incoming IP multicast packets out all interfaces that lead to the members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

IP Multicast Routing Protocols

The software supports the following protocols to implement IP multicast routing:

- IGMP is used between hosts on a LAN and the routers on that LAN to track the multicast groups of which hosts are members.
- Protocol Independent Multicast (PIM) is used between routers so that they can track which multicast packets to forward to each other and to their directly connected LANs.

This figure shows where these protocols operate within the IP multicast environment.

Multicast Group Transmission Scheme

IP communication consists of hosts that act as senders and receivers of traffic as shown in the first figure. Senders are called sources. Traditional IP communication is accomplished by a single host source sending packets to another single host (unicast transmission) or to all hosts (broadcast transmission). IP multicast

provides a third scheme, allowing a host to send packets to a subset of all hosts (multicast transmission). This subset of receiving hosts is called a multicast group. The hosts that belong to a multicast group are called group members.

Multicast is based on this group concept. A multicast group is an arbitrary number of receivers that join a group in order to receive a particular data stream. This multicast group has no physical or geographical boundaries--the hosts can be located anywhere on the Internet or on any private internetwork. Hosts that are interested in receiving data from a source to a particular group must join that group. Joining a group is accomplished by a host receiver by way of the Internet Group Management Protocol (IGMP).

In a multicast environment, any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group can receive packets sent to that group. Multicast packets are delivered to a group using best-effort reliability, just like IP unicast packets.



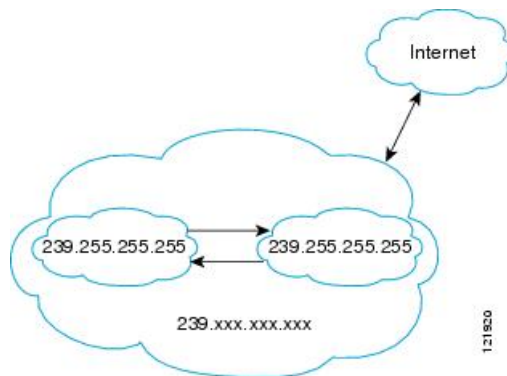
In the next figure, the receivers (the designated multicast group) are interested in receiving the video data stream from the source. The receivers indicate their interest by sending an IGMP host report to the routers in the network. The routers are then responsible for delivering the data from the source to the receivers. The routers use Protocol Independent Multicast (PIM) to dynamically create a multicast distribution tree. The video data stream will then be delivered only to the network segments that are in the path between the source and the receivers.



IP Multicast Boundary

As shown in the figure, address scoping defines domain boundaries so that domains with RPs that have the same IP address do not leak into each other. Scoping is performed on the subnet boundaries within large domains and on the boundaries between the domain and the Internet.

Figure 2: Address Scoping at Boundaries



You can set up an administratively scoped boundary on an interface for multicast group addresses using the **ip multicast boundary** command with the *access-list* argument. A standard access list defines the range of addresses affected. When a boundary is set up, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The Internet Assigned Numbers Authority (IANA) has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively scoped addresses. This range of addresses can be reused in domains administered by different organizations. They would be considered local, not globally unique.

You can configure the **filter-autorp** keyword to examine and filter Auto-RP discovery and announcement messages at the administratively scoped boundary. Any Auto-RP group range announcements from the Auto-RP packets that are denied by the boundary access control list (ACL) are removed. An Auto-RP group range announcement is permitted and passed by the boundary only if all addresses in the Auto-RP group range are permitted by the boundary ACL. If any address is not permitted, the entire group range is filtered and removed from the Auto-RP message before the Auto-RP message is forwarded.

IP Multicast Group Addressing

A multicast group is identified by its multicast group address. Multicast packets are delivered to that multicast group address. Unlike unicast addresses that uniquely identify a single host, multicast IP addresses do not identify a particular host. To receive the data sent to a multicast address, a host must join the group that address identifies. The data is sent to the multicast address and received by all the hosts that have joined the group indicating that they wish to receive traffic sent to that group. The multicast group address is assigned to a group at the source. Network administrators who assign multicast group addresses must make sure the addresses conform to the multicast address range assignments reserved by the Internet Assigned Numbers Authority (IANA).

IP Class D Addresses

IP multicast addresses have been assigned to the IPv4 Class D address space by IANA. The high-order four bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 to 239.255.255.255. A multicast address is chosen at the source (sender) for the receivers in a multicast group.



Note The Class D address range is used only for the group address or destination address of IP multicast traffic. The source address for multicast datagrams is always the unicast source address.

IP Multicast Address Scoping

The multicast address range is subdivided to provide predictable behavior for various address ranges and for address reuse within smaller domains. The table provides a summary of the multicast address ranges. A brief summary description of each range follows.

Table 11: Multicast Address Range Assignments

Name	Range	Description
Reserved Link-Local Addresses	224.0.0.0 to 224.0.0.255	Reserved for use by network protocols on a local network segment.
Globally Scoped Addresses	224.0.1.0 to 238.255.255.255	Reserved to send multicast data between organizations and across the Internet.
Source Specific Multicast	232.0.0.0 to 232.255.255.255	Reserved for use with the SSM datagram delivery model where data is forwarded only to receivers that have explicitly joined the group.

Name	Range	Description
GLOP Addresses	233.0.0.0 to 233.255.255.255	Reserved for statically defined addresses by organizations that already have an assigned autonomous system (AS) domain number.
Limited Scope Address	239.0.0.0 to 239.255.255.255	Reserved as administratively or limited scope addresses for use in private multicast domains.

Reserved Link-Local Addresses

The IANA has reserved the range 224.0.0.0 to 224.0.0.255 for use by network protocols on a local network segment. Packets with an address in this range are local in scope and are not forwarded by IP routers. Packets with link local destination addresses are typically sent with a time-to-live (TTL) value of 1 and are not forwarded by a router.

Within this range, reserved link-local addresses provide network protocol functions for which they are reserved. Network protocols use these addresses for automatic router discovery and to communicate important routing information. For example, Open Shortest Path First (OSPF) uses the IP addresses 224.0.0.5 and 224.0.0.6 to exchange link-state information.

IANA assigns single multicast address requests for network protocols or network applications out of the 224.0.1.xxx address range. Multicast routers forward these multicast addresses.



Note All the packets with reserved link-local addresses are punted to CPU by default in the ASR 903 RSP2 Module.

Globally Scoped Addresses

Addresses in the range 224.0.1.0 to 238.255.255.255 are called globally scoped addresses. These addresses are used to send multicast data between organizations across the Internet. Some of these addresses have been reserved by IANA for use by multicast applications. For example, the IP address 224.0.1.1 is reserved for Network Time Protocol (NTP).

Source Specific Multicast Addresses

Addresses in the range 232.0.0.0/8 are reserved for Source Specific Multicast (SSM) by IANA. In Cisco IOS software, you can use the `ip pim ssm` command to configure SSM for arbitrary IP multicast addresses also. SSM is an extension of Protocol Independent Multicast (PIM) that allows for an efficient data delivery mechanism in one-to-many communications. SSM is described in the [IP Multicast Delivery Modes, on page 85](#) section.

GLOP Addresses

GLOP addressing (as proposed by RFC 2770, GLOP Addressing in 233/8) proposes that the 233.0.0.0/8 range be reserved for statically defined addresses by organizations that already have an AS number reserved. This practice is called GLOP addressing. The AS number of the domain is embedded into the second and third octets of the 233.0.0.0/8 address range. For example, AS 62010 is written in hexadecimal format as F23A. Separating the two octets F2 and 3A results in 242 and 58 in decimal format. These values result in a subnet of 233.242.58.0/24 that would be globally reserved for AS 62010 to use.

Limited Scope Addresses

The range 239.0.0.0 to 239.255.255.255 is reserved as administratively or limited scoped addresses for use in private multicast domains. These addresses are constrained to a local group or organization. Companies, universities, and other organizations can use limited scope addresses to have local multicast applications that will not be forwarded outside their domain. Routers typically are configured with filters to prevent multicast traffic in this address range from flowing outside an autonomous system (AS) or any user-defined domain. Within an AS or domain, the limited scope address range can be further subdivided so that local multicast boundaries can be defined.



Note Network administrators may use multicast addresses in this range, inside a domain, without conflicting with others elsewhere in the Internet.

Layer 2 Multicast Addresses

Historically, network interface cards (NICs) on a LAN segment could receive only packets destined for their burned-in MAC address or the broadcast MAC address. In IP multicast, several hosts need to be able to receive a single data stream with a common destination MAC address. Some means had to be devised so that multiple hosts could receive the same packet and still be able to differentiate between several multicast groups. One method to accomplish this is to map IP multicast Class D addresses directly to a MAC address. Using this method, NICs can receive packets destined to many different MAC address.

Cisco Group Management Protocol (CGMP) is used on routers connected to Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary for those Catalyst switches that cannot distinguish between IP multicast data packets and IGMP report messages, both of which are addressed to the same group address at the MAC level.

IP Multicast Delivery Modes

IP multicast delivery modes differ only for the receiver hosts, not for the source hosts. A source host sends IP multicast packets with its own IP address as the IP source address of the packet and a group address as the IP destination address of the packet.

Source Specific Multicast

Source Specific Multicast (SSM) is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is a core network technology for the Cisco implementation of IP multicast targeted for audio and video broadcast application environments.

For the SSM delivery mode, an IP multicast receiver host must use IGMP Version 3 (IGMPv3) to subscribe to channel (S,G). By subscribing to this channel, the receiver host is indicating that it wants to receive IP multicast traffic sent by source host S to group G. The network will deliver IP multicast packets from source host S to group G to all hosts in the network that have subscribed to the channel (S, G).

SSM does not require group address allocation within the network, only within each source host. Different applications running on the same source host must use different SSM groups. Different applications running on different source hosts can arbitrarily reuse SSM group addresses without causing any excess traffic on the network.



CHAPTER 11

Configuring Basic IP Multicast Routing

- [Prerequisites for Basic IP Multicast Routing, on page 87](#)
- [Restrictions for Basic IP Multicast Routing, on page 87](#)
- [Information About Basic IP Multicast Routing, on page 87](#)
- [How to Configure Basic IP Multicast Routing, on page 89](#)
- [Monitoring and Maintaining Basic IP Multicast Routing, on page 96](#)

Prerequisites for Basic IP Multicast Routing

The following are the prerequisites for configuring basic IP multicast routing:

- You must configure the PIM version and the PIM mode in order to perform IP multicast routing. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting. You can configure an interface to be in the PIM dense mode, sparse mode, or sparse-dense mode.
- Enabling PIM on an interface also enables IGMP operation on that interface. (To participate in IP multicasting, the multicast hosts, routers, and multilayer device must have IGMP operating.)

If you enable PIM on multiple interfaces, when most of these interfaces are not on the outgoing interface list, and IGMP snooping is disabled, the outgoing interface might not be able to sustain line rate for multicast traffic because of the extra replication.

Restrictions for Basic IP Multicast Routing

The following are the restrictions for IP multicast routing:

- IP Multicast routing is supported only on Catalyst 3560-CX switches.

Information About Basic IP Multicast Routing

IP multicasting is an efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address.

The sending host inserts the multicast group address into the IP destination address field of the packet, and IP multicast routers and multilayer devices forward incoming IP multicast packets out all interfaces that lead to members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

Default IP Multicast Routing Configuration

This table displays the default IP multicast routing configuration.

Table 12: Default IP Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM stub routing	None configured.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kb/s.
PIM router query message interval	30 seconds.

sdr Listener Support

The MBONE is the small subset of Internet routers and hosts that are interconnected and capable of forwarding IP multicast traffic. Other multimedia content is often broadcast over the MBONE. Before you can join a multimedia session, you need to know what multicast group address and port are being used for the session, when the session is going to be active, and what sort of applications (audio, video, and so forth) are required on your workstation. The MBONE Session Directory Version 2 (sdr) tool provides this information. This freeware application can be downloaded from several sites on the World Wide Web, one of which is <http://www.video.ja.net/mice/index.html>.

SDR is a multicast application that listens to a well-known multicast group address and port for Session Announcement Protocol (SAP) multicast packets from SAP clients, which announce their conference sessions. These SAP packets contain a session description, the time the session is active, its IP multicast group addresses, media format, contact person, and other information about the advertised multimedia session. The information in the SAP packet is displayed in the SDR Session Announcement window.

How to Configure Basic IP Multicast Routing

Configuring Basic IP Multicast Routing

By default, multicast routing is disabled, and there is no default mode setting.

This procedure is required.

Before you begin

You must configure the PIM version and the PIM mode. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic join messages are received from downstream devices or when there is a directly connected member on the interface. When forwarding from a LAN, sparse-mode operation occurs if there is an RP known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. The multicast source address must be on the directly connected incoming interface (that is part of the same subnet) of the first-hop router for both PIM dense mode and PIM any-source multicast mode. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router might send join messages toward the source to build a source-based distribution tree.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim** {dense-mode | sparse-mode | sparse-dense-mode}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	<p>Specifies the Layer 3 interface on which you want to enable multicast routing, and enters interface configuration mode.</p> <p>The specified interface must be one of the following:</p> <ul style="list-style-type: none"> • A routed port—A physical port that has been configured as a Layer 3 port by entering the no switchport interface configuration command. You will also need to enable IP PIM sparse-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. • An SVI—A VLAN interface created by using the interface vlan <i>vlan-id</i> global configuration command. You will also need to enable IP PIM sparse-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. <p>These interfaces must have IP addresses assigned to them.</p>
Step 4	<p>ip pim {dense-mode sparse-mode sparse-dense-mode}</p> <p>Example:</p> <pre>Switch(config-if)# ip pim sparse-dense-mode</pre>	<p>Enables a PIM mode on the interface.</p> <p>By default, no mode is configured.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • dense-mode—Enables dense mode of operation. • sparse-mode—Enables sparse mode of operation. If you configure sparse mode, you must also configure an RP. • sparse-dense-mode—Causes the interface to be treated in the mode in which the group belongs. Sparse-dense mode is the recommended setting. <p>Note To disable PIM on an interface, use the no ip pim interface configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Optional IP Multicast Routing Features

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **deny** *source* [*source-wildcard*]
4. **interface** *interface-id*
5. **ip multicast boundary** *access-list-number*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	access-list <i>access-list-number</i> deny <i>source</i> [<i>source-wildcard</i>] Example: <pre>Switch(config)# access-list 12 deny 224.0.1.39</pre>	Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched.

	Command or Action	Purpose
	<code>access-list 12 deny 224.0.1.40</code>	<ul style="list-style-type: none"> For <i>source</i>, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 4	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface to be configured, and enters interface configuration mode.
Step 5	ip multicast boundary <i>access-list-number</i> Example: <pre>Switch(config-if)# ip multicast boundary 12</pre>	Configures the boundary, specifying the access list you created in Step 2.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Multicast VRFs

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 2	ip routing Example: Switch(config)# <code>ip routing</code>	Enables IP routing mode.
Step 3	ip vrf <i>vrf-name</i> Example: Switch(config)# <code>ip vrf vpn1</code>	Names the VRF, and enter VRF configuration mode.
Step 4	rd <i>route-distinguisher</i> Example: Switch(config-vrf)# <code>rd 100:2</code>	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y)
Step 5	route-target {export import both} <i>route-target-ext-community</i> Example: Switch(config-vrf)# <code>route-target import 100:2</code>	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The <i>route-target-ext-community</i> should be the same as the <i>route-distinguisher</i> entered in Step 4.
Step 6	import map <i>route-map</i> Example: Switch(config-vrf)# <code>import map importmap1</code>	(Optional) Associates a route map with the VRF.
Step 7	ip multicast-routing vrf <i>vrf-name</i> distributed Example: Switch(config-vrf)# <code>ip multicast-routing vrf vpn1 distributed</code>	(Optional) Enables global multicast routing for VRF table.
Step 8	interface <i>interface-id</i> Example: Switch(config-vrf)# <code>interface gigabitethernet 1/0/2</code>	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or an SVI.
Step 9	ip vrf forwarding <i>vrf-name</i> Example: Switch(config-if)# <code>ip vrf forwarding vpn1</code>	Associates the VRF with the Layer 3 interface.

	Command or Action	Purpose
Step 10	ip address <i>ip-address</i> <i>mask</i> Example: <pre>Switch(config-if)# ip address 10.1.5.1 255.255.255.0</pre>	Configures IP address for the Layer 3 interface.
Step 11	ip pim sparse-dense mode Example: <pre>Switch(config-if)# ip pim sparse-dense mode</pre>	Enables PIM on the VRF-associated Layer 3 interface.
Step 12	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 13	show ip vrf [<i>brief</i> <i>detail</i> <i>interfaces</i>] [<i>vrf-name</i>] Example: <pre>Switch# show ip vrf detail vpn1</pre>	Verifies the configuration. Displays information about the configured VRFs.
Step 14	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Advertising Multicast Multimedia Sessions Using SAP Listener

Enable SAP listener support when you want to use session description and announcement protocols and applications to assist the advertisement of multicast multimedia conferences and other multicast sessions and to communicate the relevant session setup information to prospective participants.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sap cache-timeout** *minutes*
4. **interface** *type number*
5. **ip sap listen**
6. **end**
7. **clear ip sap** [*group-address* | “*session-name*”]
8. **show ip sap** [*group-address* | “*session-name*”] **detail**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	ip sap cache-timeout <i>minutes</i> Example: <pre>Router(config)# ip sap cache-timeout 600</pre>	(Optional) Limits how long a SAP cache entry stays active in the cache. <ul style="list-style-type: none"> • By default, SAP cache entries are deleted 24 hours after they are received from the network.
Step 4	interface <i>type number</i> Example: <pre>Router(config)# interface ethernet 1</pre>	Selects an interface that is connected to hosts on which IGMPv3 can be enabled.
Step 5	ip sap listen Example: <pre>Router(config-if)# ip sap listen</pre>	Enables the software to listen to session directory announcements.
Step 6	end Example: <pre>Router(config-if)# end</pre>	Ends the session and returns to EXEC mode.
Step 7	clear ip sap [<i>group-address</i> "<i>session-name</i>"] Example: <pre>Router# clear ip sap "Sample Session"</pre>	Deletes a SAP cache entry or the entire SAP cache.
Step 8	show ip sap [<i>group-address</i> "<i>session-name</i>"] detail Example: <pre>Router# show ip sap 224.2.197.250 detail</pre>	(Optional) Displays the SAP cache.

Monitoring and Maintaining Basic IP Multicast Routing

Clearing Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database might be necessary when the contents of the particular structure are or suspected to be invalid.

You can use any of the privileged EXEC commands in the following table to clear IP multicast caches, tables, and databases.

Table 13: Commands for Clearing Caches, Tables, and Databases

Command	Purpose
clear ip igmp group { group [<i>hostname</i> <i>IP address</i>] vrf name group [<i>hostname</i> <i>IP address</i>] }	Deletes entries from the IGMP cache.
clear ip mroute { * [<i>hostname</i> <i>IP address</i>] vrf name group [<i>hostname</i> <i>IP address</i>] }	Deletes entries from the IP multicast routing table.
clear ip sap [<i>group-address</i> “ <i>session-name</i> ”]	Deletes the Session Directories cache entry.

Displaying System and Network Statistics

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 14: Commands for Displaying System and Network Statistics

Command	Purpose
ping [<i>group-name</i> <i>group-address</i>]	Sends an ICMP Echo Request to a multicast group.
show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>type-number</i>]	Displays the multicast groups that are directly connected to the interface.
show ip igmp interface [<i>type number</i>]	Displays multicast-related information about the interface.
show ip mroute [<i>group-name</i> <i>group-address</i>] [<i>source</i>] [count interface proxy pruned summary verbose]	Displays the contents of the IP multicast routing table.
show ip pim interface [<i>type number</i>] [count detail df stats]	Displays information about interfaces configured for PIM.

Command	Purpose
show ip pim neighbor [<i>type number</i>]	Lists the PIM neighbors discovered by the device.
show ip pim rp [<i>group-name</i> <i>group-address</i>]	Displays the RP routers associated with a group of images.
show ip rpf { <i>source-address</i> <i>name</i> }	<p>Displays how the device is doing Reverse Path Forwarding (RPF) checks (in the routing table, or static mroutes).</p> <p>Command parameters include:</p> <ul style="list-style-type: none"> • <i>Host name</i> or <i>IP address</i>—IP name or address. • Select—Group-based VRF select in. • vrf—Selects VPN Routing/Forwarding table.
show ip sap [<i>group</i> " <i>session-name</i> " detail]	<p>Displays the Session Announcement Protocol (SAP) sessions.</p> <p>Command parameters include:</p> <ul style="list-style-type: none"> • <i>A.B.C.D</i>—IP group address. • <i>WORD</i>—Session name (in double quotes). • detail—Session details.



CHAPTER 12

Configuring IGMP

- [Prerequisites for IGMP, on page 99](#)
- [Restrictions for Configuring IGMP, on page 99](#)
- [Information About IGMP, on page 100](#)
- [How to Configure IGMP, on page 105](#)
- [Monitoring IGMP, on page 115](#)
- [Configuration Examples for IGMP, on page 116](#)

Prerequisites for IGMP

- Before performing the tasks in this module, you should be familiar with the concepts explained in the "IP Multicast Routing Technology Overview" module.
- The tasks in this module assume that IP multicast has been enabled and that the Protocol Independent Multicast (PIM) interfaces have been configured using the tasks described in the "Configuring IP Multicast Routing" module.

Restrictions for Configuring IGMP

The following are the restrictions for configuring IGMP:

- The device supports IGMP Versions 1, 2, and 3.



Note For IGMP Version 3, only IGMP Version 3 BISS (Basic IGMPv3 Snooping Support) is supported.

- IGMP Version 3 uses new membership report messages that might not be correctly recognized by older IGMP snooping devices.
- IGMPv3 can operate with both ISM and SSM. In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.

Information About IGMP

Role of the Internet Group Management Protocol

IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN. Enabling PIM on an interface also enables IGMP. IGMP provides a means to automatically control and limit the flow of multicast traffic throughout your network with the use of special multicast queriers and hosts.

- A querier is a network device, such as a router, that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver, including routers, that sends report messages (in response to query messages) to inform the querier of a host membership. Hosts use IGMP messages to join and leave multicast groups.

Hosts identify group memberships by sending IGMP messages to their local multicast device. Under IGMP, devices listen to IGMP messages and periodically send out queries to discover which groups are active or inactive on a particular subnet.

IGMP Multicast Addresses

IP multicast traffic uses group addresses, which are Class D IP addresses. The high-order four bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 to 239.255.255.255.

Multicast addresses in the range 224.0.0.0 to 224.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are transmitted using IP multicast group addresses as follows:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the device is querying.
- IGMP group membership reports are destined to the group IP address for which the device is reporting.
- IGMPv2 leave-group messages are destined to the address 224.0.0.2 (all devices on a subnet).
- IGMPv3 membership reports are destined to the address 224.0.0.22; all IGMPv3-capable multicast devices must listen to this address.

IGMP Versions

The device supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the device. For example, if IGMP snooping is enabled and the querier's version is IGMPv2, and the device receives an IGMPv3 report from a host, then the device can forward the IGMPv3 report to the multicast router.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMP Version 1

IGMP version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer device to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.



Note IGMP version 2 is the default version for the device.

IGMP Version 3

The device supports IGMP version 3.

An IGMPv3 device supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMPv3 Host Signalling

In IGMPv3, hosts signal membership to last hop routers of multicast groups. Hosts can signal group membership with filtering capabilities with respect to sources. A host can either signal that it wants to receive traffic from all sources sending to a group except for some specific sources (called exclude mode), or that it wants to receive traffic only from some specific sources sending to the group (called include mode).

IGMPv3 can operate with both Internet Standard Multicast (ISM) and Source Specific Multicast (SSM). In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.

IGMP Versions Differences

There are three versions of IGMP, as defined by Request for Comments (RFC) documents of the Internet Engineering Task Force (IETF). IGMPv2 improves over IGMPv1 by adding the ability for a host to signal desire to leave a multicast group and IGMPv3 improves over IGMPv2 mainly by adding the ability to listen to multicast originating from a set of source IP addresses only.

Table 15: IGMP Versions

IGMP Version	Description
IGMPv1	Provides the basic query-response mechanism that allows the multicast device to determine which multicast groups are active and other processes that enable hosts to join and leave a multicast group. RFC 1112 defines the IGMPv1 host extensions for IP multicasting.
IGMPv2	Extends IGMP, allowing such capabilities as the IGMP leave process, group-specific queries, and an explicit maximum response time field. IGMPv2 also adds the capability for devices to elect the IGMP querier without dependence on the multicast protocol to perform this task. RFC 2236 defines IGMPv2.



Note By default, enabling a PIM on an interface enables IGMPv2 on that device. IGMPv2 was designed to be as backward compatible with IGMPv1 as possible. To accomplish this backward compatibility, RFC 2236 defined special interoperability rules. If your network contains legacy IGMPv1 hosts, you should be familiar with these operability rules. For more information about IGMPv1 and IGMPv2 interoperability, see RFC 2236, Internet Group Management Protocol, Version 2 .

Devices That Run IGMPv1

IGMPv1 devices send IGMP queries to the “all-hosts” multicast address of 224.0.0.1 to solicit multicast groups with active multicast receivers. The multicast receivers also can send IGMP reports to the device to notify it that they are interested in receiving a particular multicast stream. Hosts can send the report asynchronously or in response to the IGMP queries sent by the device. If more than one multicast receiver exists for the same multicast group, only one of these hosts sends an IGMP report message; the other hosts suppress their report messages.

In IGMPv1, there is no election of an IGMP querier. If more than one device on the segment exists, all the devices send periodic IGMP queries. IGMPv1 has no special mechanism by which the hosts can leave the group. If the hosts are no longer interested in receiving multicast packets for a particular group, they simply do not reply to the IGMP query packets sent from the device. The device continues sending query packets. If the device does not hear a response in three IGMP queries, the group times out and the device stops sending multicast packets on the segment for the group. If the host later wants to receive multicast packets after the timeout period, the host simply sends a new IGMP join to the device, and the device begins to forward the multicast packet again.

If there are multiple devices on a LAN, a designated router (DR) must be elected to avoid duplicating multicast traffic for connected hosts. PIM devices follow an election process to select a DR. The PIM device with the highest IP address becomes the DR.

The DR is responsible for the following tasks:

- Sending PIM register and PIM Join and Prune messages toward the rendezvous point (RP) to inform it about host group membership.
- Sending IGMP host-query messages.
- Sending host-query messages by default every 60 seconds in order to keep the IGMP overhead on hosts and networks very low.

Devices That Run IGMPv2

IGMPv2 improves the query messaging capabilities of IGMPv1.

The query and membership report messages in IGMPv2 are identical to the IGMPv1 messages with two exceptions:

- IGMPv2 query messages are broken into two categories: general queries (identical to IGMPv1 queries) and group-specific queries.
- IGMPv1 membership reports and IGMPv2 membership reports have different IGMP type codes.

IGMPv2 also enhances IGMP by providing support for the following capabilities:

- Querier election process--Provides the capability for IGMPv2 devices to elect the IGMP querier without having to rely on the multicast routing protocol to perform the process.
- Maximum Response Time field--A new field in query messages permits the IGMP querier to specify the maximum query-response time. This field permits the tuning of the query-response process to control response burstiness and to fine-tune leave latencies.
- Group-Specific Query messages--Permits the IGMP querier to perform the query operation on a specific group instead of all groups.
- Leave-Group messages--Provides hosts with a method of notifying devices on the network that they wish to leave the group.

Unlike IGMPv1, in which the DR and the IGMP querier are typically the same device, in IGMPv2 the two functions are decoupled. The DR and the IGMP querier are selected based on different criteria and may be different devices on the same subnet. The DR is the device with the highest IP address on the subnet, whereas the IGMP querier is the device with the lowest IP address.

Query messages are used to elect the IGMP querier as follows:

1. When IGMPv2 devices start, they each multicast a general query message to the all-systems group address of 224.0.0.1 with their interface address in the source IP address field of the message.
2. When an IGMPv2 device receives a general query message, the device compares the source IP address in the message with its own interface address. The device with the lowest IP address on the subnet is elected the IGMP querier.
3. All devices (excluding the querier) start the query timer, which is reset whenever a general query message is received from the IGMP querier. If the query timer expires, it is assumed that the IGMP querier has gone down, and the election process is performed again to elect a new IGMP querier.

By default, the timer is two times the query interval.

IGMP Join and Leave Process

IGMP Join Process

When a host wants to join a multicast group, the host sends one or more unsolicited membership reports for the multicast group it wants to join. The IGMP join process is the same for IGMPv1 and IGMPv2 hosts.

In IGMPv3, the join process for hosts proceeds as follows:

- When a host wants to join a group, it sends an IGMPv3 membership report to 224.0.0.22 with an empty EXCLUDE list.
- When a host wants to join a specific channel, it sends an IGMPv3 membership report to 224.0.0.22 with the address of the specific source included in the INCLUDE list.
- When a host wants to join a group excluding particular sources, it sends an IGMPv3 membership report to 224.0.0.22 excluding those sources in the EXCLUDE list.



Note If some IGMPv3 hosts on a LAN wish to exclude a source and others wish to include the source, then the device will send traffic for the source on the LAN (that is, inclusion trumps exclusion in this situation).

IGMP Leave Process

The method that hosts use to leave a group varies depending on the version of IGMP in operation.

IGMPv1 Leave Process

There is no leave-group message in IGMPv1 to notify the devices on the subnet that a host no longer wants to receive the multicast traffic from a specific group. The host simply stops processing traffic for the multicast group and ceases responding to IGMP queries with IGMP membership reports for the group. As a result, the only way IGMPv1 devices know that there are no longer any active receivers for a particular multicast group on a subnet is when the devices stop receiving membership reports. To facilitate this process, IGMPv1 devices associate a countdown timer with an IGMP group on a subnet. When a membership report is received for the group on the subnet, the timer is reset. For IGMPv1 devices, this timeout interval is typically three times the query interval (3 minutes). This timeout interval means that the device may continue to forward multicast traffic onto the subnet for up to 3 minutes after all hosts have left the multicast group.

IGMPv2 Leave Process

IGMPv2 incorporates a leave-group message that provides the means for a host to indicate that it wishes to stop receiving multicast traffic for a specific group. When an IGMPv2 host leaves a multicast group, if it was the last host to respond to a query with a membership report for that group, it sends a leave-group message to the all-devices multicast group (224.0.0.2).

IGMPv3 Leave Process

IGMPv3 enhances the leave process by introducing the capability for a host to stop receiving traffic from a particular group, source, or channel in IGMP by including or excluding sources, groups, or channels in IGMPv3 membership reports.

Default IGMP Configuration

This table displays the default IGMP configuration for the device.

Table 16: Default IGMP Configuration

Feature	Default Setting
Multilayer device as a member of a multicast group	No group memberships are defined.

Feature	Default Setting
Access to multicast groups	All groups are allowed on an interface.
IGMP version	Version 2 on all interfaces.
IGMP host-query message interval	60 seconds on all interfaces.
IGMP query timeout	60 seconds on all interfaces.
IGMP maximum query response time	10 seconds on all interfaces.
Multilayer device as a statically connected member	Disabled.

How to Configure IGMP

Configuring the Device as a Member of a Group

You can configure the device as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer devices that you administer are members of a multicast group, pinging that group causes all of these devices to respond. The devices respond to ICMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.



Caution Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp join-group** *group-address*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 4	ip igmp join-group <i>group-address</i> Example: Switch(config-if)# ip igmp join-group 225.2.2.2	Configures the device to join a multicast group. By default, no group memberships are defined. For <i>group-address</i> , specify the multicast IP address in dotted decimal notation.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Switch# show ip igmp interface	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Controlling Access to IP Multicast Group

The switch sends IGMP host-query messages to find which multicast groups have members on attached local networks. The switch then forwards to these group members all packets addressed to the multicast group. You can place a filter on each interface to restrict the multicast groups that hosts on the subnet serviced by the interface can join.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp access-group** *access-list-number*
5. **exit**
6. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
7. **end**
8. **show ip igmp interface** [*interface-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface GigabitEthernet 1/0/12</pre>	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip igmp access-group <i>access-list-number</i> Example: <pre>Switch(config-if)# ip igmp access-group 10</pre>	Specifies the multicast groups that hosts on the subnet serviced by an interface can join. By default, all groups are allowed on an interface. For <i>access-list-number</i> , specify an IP standard access list number. The range is 1 to 199. Note To disable groups on an interface, use the no ip igmp access-group interface configuration command.
Step 5	exit Example: <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.

	Command or Action	Purpose
Step 6	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 10 permit</pre>	<p>Creates a standard access list.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, specify the access list created in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, specify the multicast group that hosts on the subnet can join. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Switch(config-igmp-profile)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show ip igmp interface [<i>interface-id</i>]</p> <p>Example:</p> <pre>Switch# show ip igmp interface</pre>	Verifies your entries.

Changing the IGMP Version

By default, the switch uses IGMP Version 2, which provides features such as the IGMP query timeout and the maximum query response time.

All systems on the subnet must support the same version. The switch does not automatically detect Version 1 systems and switch to Version 1. You can mix Version 1 and Version 2 hosts on the subnet because Version 2 routers or switches always work correctly with IGMPv1 hosts.

Configure the switch for Version 1 if your hosts do not support Version 2.

This procedure is optional.

SUMMARY STEPS

- enable**
- configure terminal**
- interface** *interface-id*
- ip igmp version** {1 | 2 | 3 }

5. `end`
6. `show ip igmp interface [interface-id]`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>interface interface-id</code></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	<p>Specifies the interface to be configured, and enters the interface configuration mode.</p>
Step 4	<p><code>ip igmp version {1 2 3 }</code></p> <p>Example:</p> <pre>Switch(config-if)# ip igmp version 2</pre>	<p>Specifies the IGMP version that the switch uses.</p> <p>Note If you change to Version 1, you cannot configure the <code>ip igmp query-interval</code> or the <code>ip igmp query-max-response-time</code> interface configuration commands.</p> <p>To return to the default setting, use the <code>no ip igmp version</code> interface configuration command.</p>
Step 5	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p><code>show ip igmp interface [interface-id]</code></p> <p>Example:</p> <pre>Switch# show ip igmp interface</pre>	<p>Verifies your entries.</p>
Step 7	<p><code>copy running-config startup-config</code></p> <p>Example:</p>	<p>(Optional) Saves your entries in the configuration file.</p>

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Modifying the IGMP Host-Query Message Interval

The device periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The device sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The device elects a PIM designated router (DR) for the LAN (subnet). The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router. With IGMPv2, the DR is the router or multilayer device with the highest IP address. With IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `ip igmp query-interval seconds`
5. `end`
6. `show ip igmp interface [interface-id]`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface interface-id Example:	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.

	Command or Action	Purpose
	Switch(config)# interface gigabitethernet 1/0/1	
Step 4	ip igmp query-interval <i>seconds</i> Example: Switch(config-if)# ip igmp query-interval 75	Configures the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Switch# show ip igmp interface	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Changing the IGMP Query Timeout for IGMPv2

If you are using IGMPv2, you can specify the period of time before the device takes over as the querier for the interface. By default, the device waits twice the query interval period controlled by the **ip igmp query-interval** interface configuration command. After that time, if the device has received no queries, it becomes the querier.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp querier-timeout** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 4	ip igmp querier-timeout <i>seconds</i> Example: Switch(config-if)# ip igmp querier-timeout 120	Specifies the IGMP query timeout. The default is 60 seconds (twice the query interval). The range is 60 to 300.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Switch# show ip igmp interface	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the device to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the device to prune groups faster.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp query-max-response-time** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 4	ip igmp query-max-response-time <i>seconds</i> Example: <pre>Switch(config-if)# ip igmp query-max-response-time 15</pre>	Changes the maximum query response time advertised in IGMP queries. The default is 10 seconds. The range is 1 to 25.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Switch# show ip igmp interface	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Device as a Statically Connected Member

At various times, either there is not a group member on a network segment or a host that cannot report its group membership by using IGMP. However, you may want multicast traffic to be sent to that network segment. The following commands are used to pull multicast traffic down to a network segment:

- **ip igmp join-group**—The device accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the device from fast switching.
- **ip igmp static-group**—The device does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the device itself is not a member, as evidenced by lack of an L (local) flag in the multicast route entry.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp static-group** *group-address*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> <code>enable</code>	
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 4	ip igmp static-group <i>group-address</i> Example: Switch(config-if)# <code>ip igmp static-group 239.100.100.101</code>	Configures the device as a statically connected member of a group. By default, this feature is disabled.
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Switch# <code>show ip igmp interface gigabitethernet 1/0/1</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring IGMP

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 17: Commands for Displaying System and Network Statistics

Command	Purpose
<code>show ip igmp groups [type-number detail]</code>	Displays the multicast groups that are directed to the interface. Displays IGMP.
<code>show ip igmp interface [type number]</code>	Displays multicast-related information about the interface.
<code>show ip igmp profile [profile_number]</code>	Displays IGMP profile information.
<code>show ip igmp ssm-mapping [hostname/IP address]</code>	Displays IGMP SSM mapping information.
<code>show ip igmp static-group {class-map [interface [type]]}</code>	Displays static group information.
<code>show ip igmp vrf</code>	Displays the selected VPN routing/forwarding table.

Configuration Examples for IGMP

Example: Configuring the Device as a Member of a Multicast Group

This example shows how to enable the device to join multicast group 255.2.2.2:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip igmp join-group 255.2.2.2
Switch(config-if)#
```

Example: Controlling Access to IP Multicast Groups

This example shows how to configure hosts attached to a port as able to join only group 255.2.2.2:

```
Switch(config)# access-list 1 255.2.2.2 0.0.0.0
Switch(config-if)# interface gigabitethernet1/0/1
Switch(config-if)# ip igmp access-group 1
```



CHAPTER 13

Configuring IGMP Snooping and Multicast VLAN Registration

- [Prerequisites for Configuring IGMP Snooping and MVR, on page 117](#)
- [Restrictions for Configuring IGMP Snooping and MVR, on page 118](#)
- [Information About IGMP Snooping and MVR, on page 119](#)
- [How to Configure IGMP Snooping and MVR, on page 128](#)
- [Monitoring IGMP Snooping and MVR, on page 156](#)
- [Configuration Examples for IGMP Snooping and MVR, on page 158](#)

Prerequisites for Configuring IGMP Snooping and MVR

Prerequisites for IGMP Snooping

Observe these guidelines when configuring the IGMP snooping querier:

- Configure the VLAN in global configuration mode.
- Configure an IP address on the VLAN interface. When enabled, the IGMP snooping querier uses the IP address as the query source address.
- If there is no IP address configured on the VLAN interface, the IGMP snooping querier tries to use the configured global IP address for the IGMP querier. If there is no global IP address specified, the IGMP querier tries to use the VLAN device virtual interface (SVI) IP address (if one exists). If there is no SVI IP address, the device uses the first available IP address configured on the device. The first IP address available appears in the output of the **show ip interface** privileged EXEC command. The IGMP snooping querier does not generate an IGMP general query if it cannot find an available IP address on the device.
- The IGMP snooping querier supports IGMP Versions 1 and 2.
- When administratively enabled, the IGMP snooping querier moves to the nonquerier state if it detects the presence of a multicast router in the network.
- When it is administratively enabled, the IGMP snooping querier moves to the operationally disabled state under these conditions:
 - IGMP snooping is disabled in the VLAN.

- PIM is enabled on the SVI of the corresponding VLAN.

-
-

Prerequisites for MVR

The following are the prerequisites for Multicast VLAN Registration (MVR):

- To use MVR, the device must be running the LAN Base image.

Restrictions for Configuring IGMP Snooping and MVR

Restrictions for IGMP Snooping

The following are the restrictions for IGMP snooping:

- The switch supports homogeneous stacking and mixed stacking. Mixed stacking is supported only with the Catalyst 2960-S switches. A homogenous stack can have up to eight stack members, while a mixed stack can have up to four stack members. All switches in a switch stack must be running the LAN Base image.
- IGMPv3 join and leave messages are not supported on devices running IGMP filtering or Multicast VLAN registration (MVR).
- IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.
- The IGMP configurable leave time is only supported on hosts running IGMP Version 2. IGMP version 2 is the default version for the device.

The actual leave latency in the network is usually the configured leave time. However, the leave time might vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

- The IGMP throttling action restriction can be applied only to Layer 2 ports. You can use **ip igmp max-groups action replace** interface configuration command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action {deny | replace}** command has no effect.

If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.

Restrictions for MVR

The following are restrictions for MVR:

- Only Layer 2 ports participate in MVR. You must configure ports as MVR receiver ports.

- Only one MVR multicast VLAN per device or device stack is supported.
- Receiver ports can only be access ports; they cannot be trunk ports. Receiver ports on a device can be in different VLANs, but should not belong to the multicast VLAN.
- The maximum number of multicast entries (MVR group addresses) that can be configured on a device (that is, the maximum number of television channels that can be received) is 256.
- MVR multicast data received in the source VLAN and leaving from receiver ports has its time-to-live (TTL) decremented by 1 in the device.
- Because MVR on the device uses IP multicast addresses instead of MAC multicast addresses, alias IP multicast addresses are allowed on the device. However, if the device is interoperating with Catalyst 3550 or Catalyst 3500 XL devices, you should not configure IP addresses that alias between themselves or with the reserved IP multicast addresses (in the range 224.0.0.xxx).
- Do not configure MVR on private VLAN ports.
- MVR is not supported when multicast routing is enabled on a device. If you enable multicast routing and a multicast routing protocol while MVR is enabled, MVR is disabled, and you receive a warning message. If you try to enable MVR while multicast routing and a multicast routing protocol are enabled, the operation to enable MVR is cancelled, and you receive an error message.
- MVR data received on an MVR receiver port is not forwarded to MVR source ports.
- MVR does not support IGMPv3 messages.
- The switch supports homogeneous stacking and mixed stacking. Mixed stacking is supported only with the Catalyst 2960-S switches. A homogenous stack can have up to eight stack members, while a mixed stack can have up to four stack members. All switches in a switch stack must be running the LAN Base image.

Information About IGMP Snooping and MVR

IGMP Snooping

Layer 2 devices can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN device to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the device receives an IGMP report from a host for a particular multicast group, the device adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.



Note For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The device creates one entry per VLAN

in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

The device supports IP multicast group-based bridging, instead of MAC-addressed based groups. With multicast MAC address-based groups, if an IP address being configured translates (aliases) to a previously configured MAC address or to any reserved multicast MAC addresses (in the range 224.0.0.xxx), the command fails. Because the device uses IP multicast groups, there are no address aliasing issues.

The IP multicast groups learned through IGMP snooping are dynamic. However, you can statically configure multicast groups by using the **ip igmp snooping vlan *vlan-id* static *ip_address* interface *interface-id*** global configuration command. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

You can configure an IGMP snooping querier to support IGMP snooping in subnets without multicast interfaces because the multicast traffic does not need to be routed.

If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

These sections describe IGMP snooping characteristics:

IGMP Versions

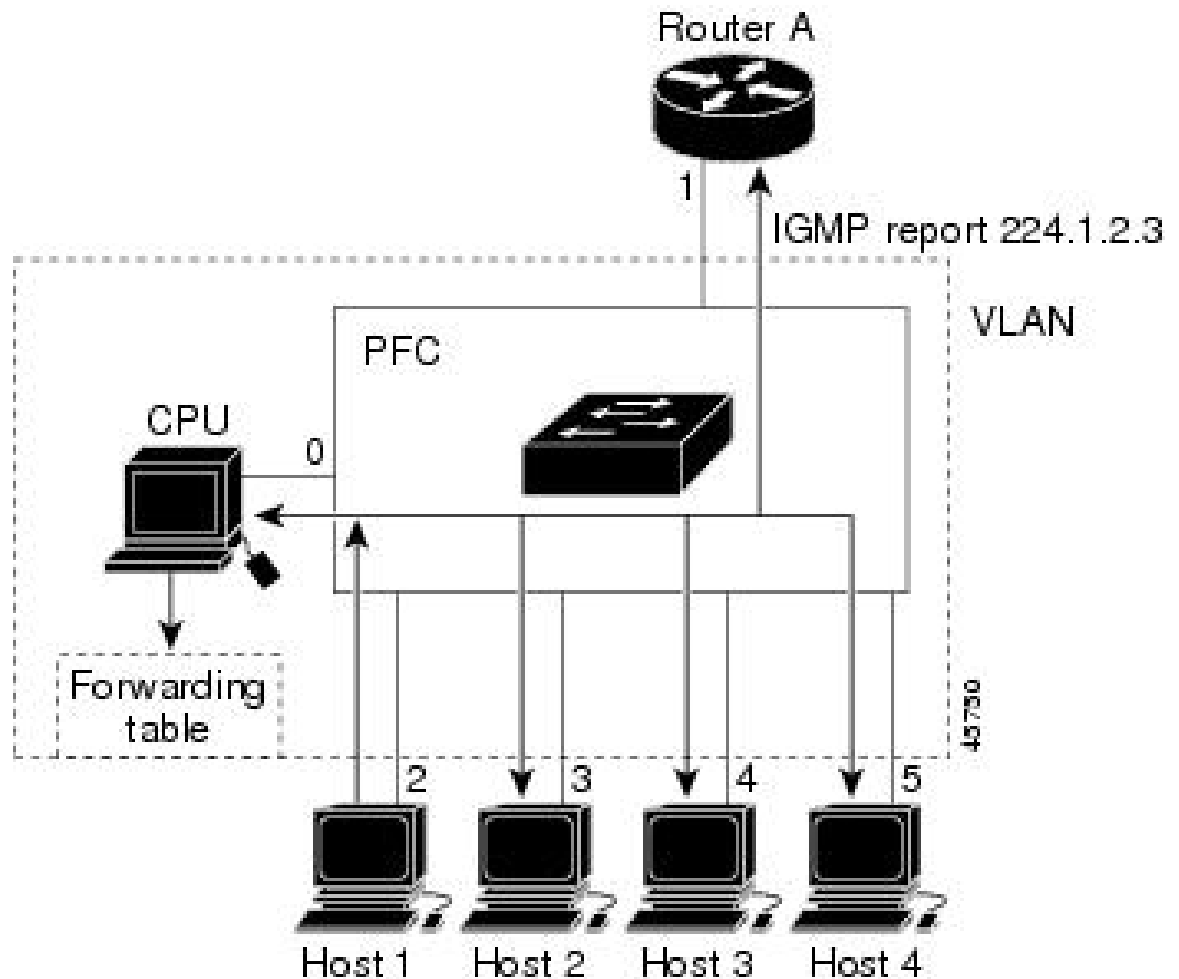
The device supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the device. For example, if IGMP snooping is enabled and the querier's version is IGMPv2, and the device receives an IGMPv3 report from a host, then the device can forward the IGMPv3 report to the multicast router.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

Joining a Multicast Group

Figure 3: Initial IGMP Join Message

When a host connected to the device wants to join an IP multicast group and it is an IGMP version 2 client, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the device receives a general query from the router, it forwards the query to all ports in the VLAN. IGMP version 1 or version 2 hosts wanting to join the multicast group respond by sending a join message to the device. The device CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group.



Router A sends a general query to the device, which forwards the query to ports 2 through 5, all of which are members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP membership report (IGMP join message) to the group. The device CPU uses the information in the IGMP report to set up a forwarding-table entry that includes the port numbers connected to Host 1 and to the router.

Table 18: IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2

The device hardware can distinguish IGMP information packets from other packets for the multicast group. The information in the table tells the switching engine to send frames addressed to the 224.1.2.3 multicast IP address that are not IGMP packets to the router and to the host that has joined the group.

Figure 4: Second Host Joining a Multicast Group

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group, the CPU receives that message and adds the port number of Host 4 to the forwarding table. Because the forwarding table directs IGMP messages only to the CPU, the message is not flooded to other ports on the device. Any

known multicast traffic is forwarded to the group and not to the CPU.

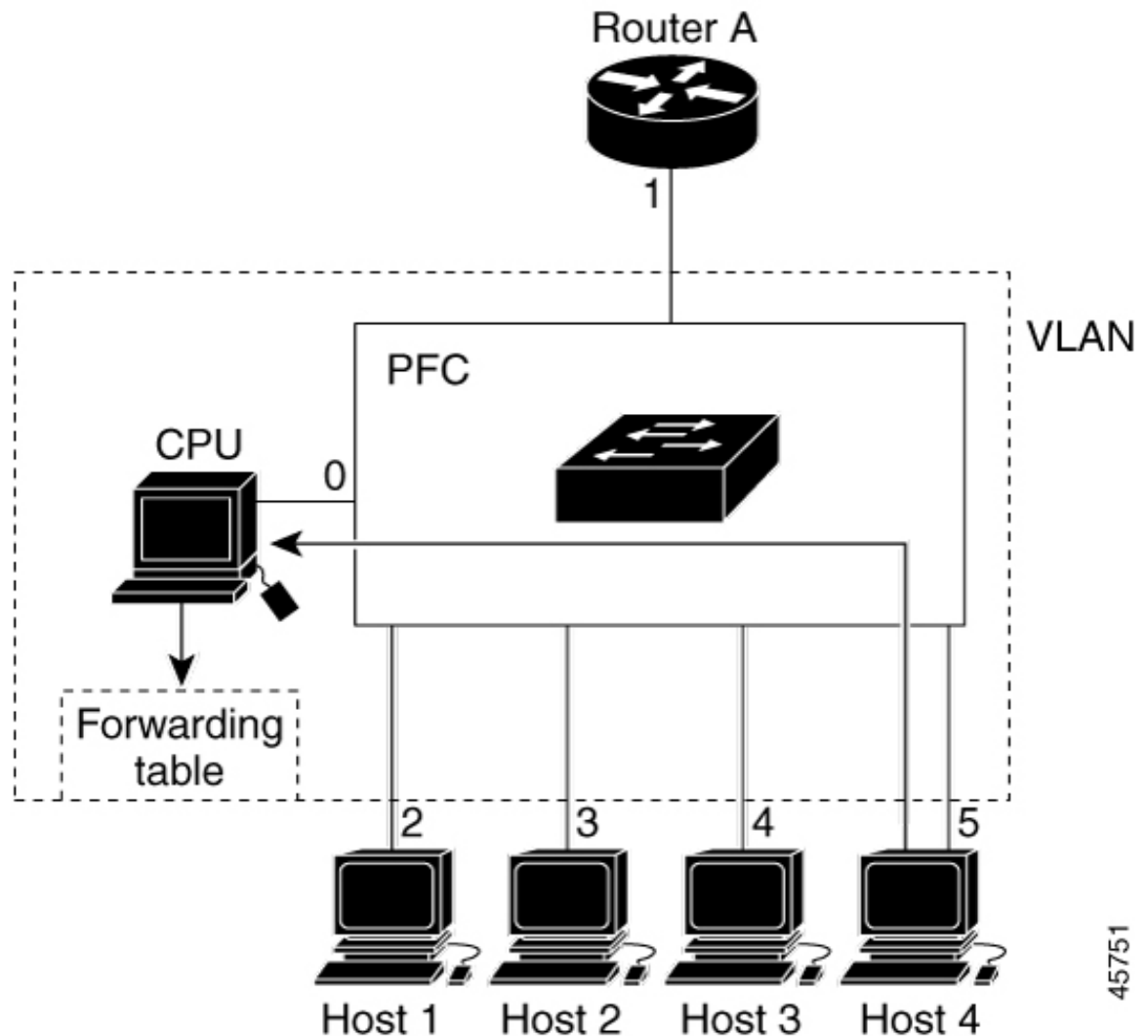


Table 19: Updated IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2, 5

Leaving a Multicast Group

The router sends periodic multicast general queries, and the device forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wants to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The device forwards multicast group traffic only to those hosts listed in the forwarding table for that IP multicast group maintained by IGMP snooping.

When hosts want to leave a multicast group, they can silently leave, or they can send a leave message. When the device receives a leave message from a host, it sends a group-specific query to learn if any other devices

connected to that interface are interested in traffic for the specific multicast group. The device then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate Leave

The device uses IGMP snooping Immediate Leave to remove from the forwarding table an interface that sends a leave message without the device sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate Leave ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Immediate Leave is only supported on IGMP version 2 hosts. IGMP version 2 is the default version for the device.



Note You should use the Immediate Leave feature only on VLANs where a single host is connected to each port. If Immediate Leave is enabled on VLANs where more than one host is connected to a port, some hosts may be dropped inadvertently.

IGMP Configurable-Leave Timer

You can configure the time that the device waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 32767 milliseconds.

IGMP Report Suppression



Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The device uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP report suppression is enabled (the default), the device sends the first IGMP report from all hosts for a group to all the multicast routers. The device does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the device forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the device forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers.

Default IGMP Snooping Configuration

This table displays the default IGMP snooping configuration for the device.

Table 20: Default IGMP Snooping Configuration

Feature	Default Setting
IGMP snooping	Enabled globally and per VLAN
Multicast routers	None configured
IGMP snooping Immediate Leave	Disabled
Static groups	None configured
TCN ¹ flood query count	2
TCN query solicitation	Disabled
IGMP snooping querier	Disabled
IGMP report suppression	Enabled

¹ (1) TCN = Topology Change Notification

Multicast VLAN Registration

Multicast VLAN Registration (MVR) is designed for applications using wide-scale deployment of multicast traffic across an Ethernet ring-based service-provider network (for example, the broadcast of multiple television channels over a service-provider network). MVR allows a subscriber on a port to subscribe and unsubscribe to a multicast stream on the network-wide multicast VLAN. It allows the single multicast VLAN to be shared in the network while subscribers remain in separate VLANs. MVR provides the ability to continuously send multicast streams in the multicast VLAN, but to isolate the streams from the subscriber VLANs for bandwidth and security reasons.

These sections describe MVR:

MVR and IGMP



Note MVR can coexist with IGMP snooping on a device.

MVR assumes that subscriber ports subscribe and unsubscribe (join and leave) these multicast streams by sending out IGMP join and leave messages. These messages can originate from an IGMP version-2-compatible host with an Ethernet connection. Although MVR operates on the underlying method of IGMP snooping, the two features operate independently of each other. One can be enabled or disabled without affecting the behavior of the other feature. However, if IGMP snooping and MVR are both enabled, MVR reacts only to join and leave messages from multicast groups configured under MVR. Join and leave messages from all other multicast groups are managed by IGMP snooping.

The device CPU identifies the MVR IP multicast streams and their associated IP multicast group in the device forwarding table, intercepts the IGMP messages, and modifies the forwarding table to include or remove the

subscriber as a receiver of the multicast stream, even though the receivers might be in a different VLAN from the source. This forwarding behavior selectively allows traffic to cross between different VLANs.

Modes of Operation

You can set the device for compatible or dynamic mode of MVR operation:

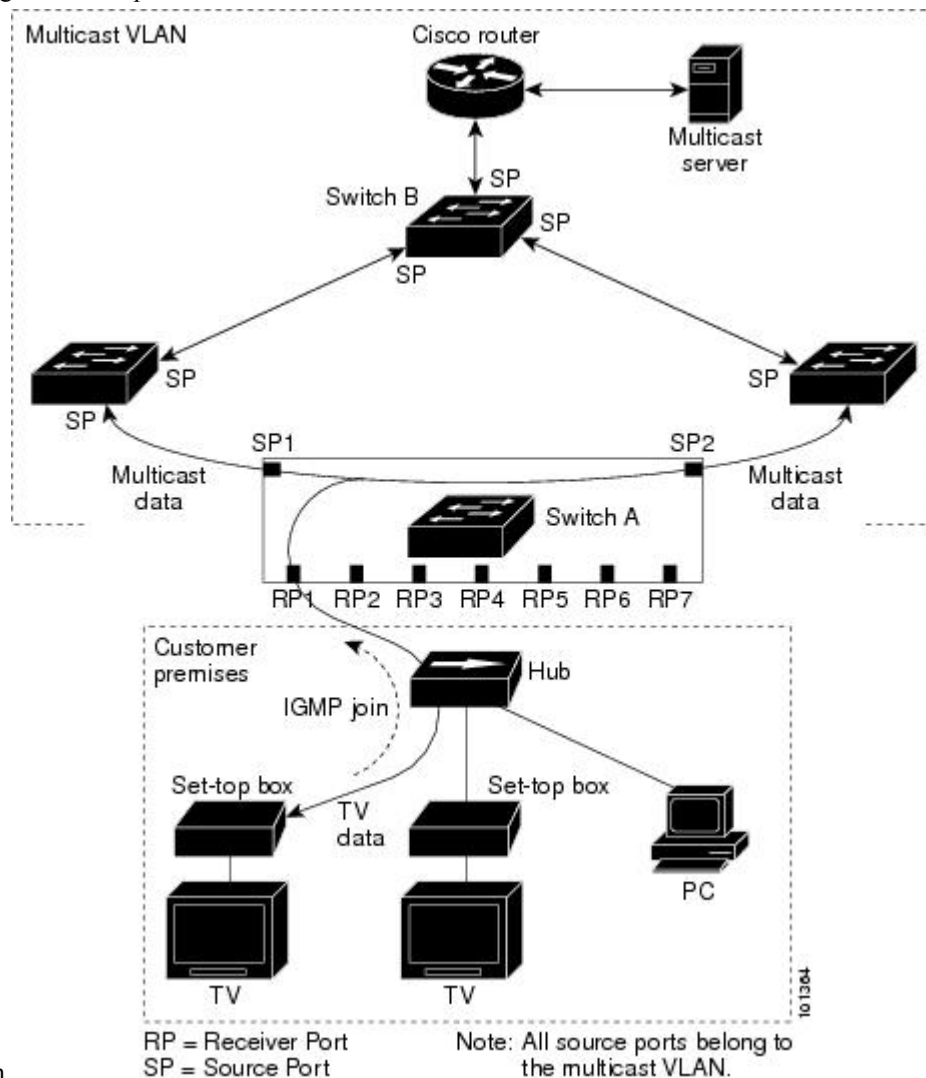
- In compatible mode, multicast data received by MVR hosts is forwarded to all MVR data ports, regardless of MVR host membership on those ports. The multicast data is forwarded only to those receiver ports that MVR hosts have joined, either by IGMP reports or by MVR static configuration. IGMP reports received from MVR hosts are never forwarded from MVR data ports that were configured in the device.
- In dynamic mode, multicast data received by MVR hosts on the device is forwarded from only those MVR data and client ports that the MVR hosts have joined, either by IGMP reports or by MVR static configuration. Any IGMP reports received from MVR hosts are also forwarded from all the MVR data ports in the host. This eliminates using unnecessary bandwidth on MVR data port links, which occurs when the device runs in compatible mode.

MVR in a Multicast Television Application

In a multicast television application, a PC or a television with a set-top box can receive the multicast stream. Multiple set-top boxes or PCs can be connected to one subscriber port, which is a device port configured as an MVR receiver port.

Figure 5: Multicast VLAN Registration Example

The following is an example



configuration.

In this example configuration, DHCP assigns an IP address to the set-top box or the PC. When a subscriber selects a channel, the set-top box or PC sends an IGMP report to Switch A to join the appropriate multicast. If the IGMP report matches one of the configured IP multicast group addresses, the device CPU modifies the hardware address table to include this receiver port and VLAN as a forwarding destination of the specified multicast stream when it is received from the multicast VLAN. Uplink ports that send and receive multicast data to and from the multicast VLAN are called MVR source ports.

When a subscriber changes channels or turns off the television, the set-top box sends an IGMP leave message for the multicast stream. The device CPU sends a MAC-based general query through the receiver port VLAN. If there is another set-top box in the VLAN still subscribing to this group, that set-top box must respond within the maximum response time specified in the query. If the CPU does not receive a response, it eliminates the receiver port as a forwarding destination for this group.

Without Immediate Leave, when the device receives an IGMP leave message from a subscriber on a receiver port, it sends out an IGMP query on that port and waits for IGMP group membership reports. If no reports

are received in a configured time period, the receiver port is removed from multicast group membership. With Immediate Leave, an IGMP query is not sent from the receiver port on which the IGMP leave was received. As soon as the leave message is received, the receiver port is removed from multicast group membership, which speeds up leave latency. Enable the Immediate-Leave feature only on receiver ports to which a single receiver device is connected.

MVR eliminates the need to duplicate television-channel multicast traffic for subscribers in each VLAN. Multicast traffic for all channels is only sent around the VLAN trunk once—only on the multicast VLAN. The IGMP leave and join messages are in the VLAN to which the subscriber port is assigned. These messages dynamically register for streams of multicast traffic in the multicast VLAN on the Layer 3 device. The access layer device, Switch A, modifies the forwarding behavior to allow the traffic to be forwarded from the multicast VLAN to the subscriber port in a different VLAN, selectively allowing traffic to cross between two VLANs.

IGMP reports are sent to the same IP multicast group address as the multicast data. The Switch A CPU must capture all IGMP join and leave messages from receiver ports and forward them to the multicast VLAN of the source (uplink) port, based on the MVR mode.

Default MVR Configuration

Table 21: Default MVR Configuration

Feature	Default Setting
MVR	Disabled globally and per interface
Multicast addresses	None configured
Query response time	0.5 second
Multicast VLAN	VLAN 1
Mode	Compatible
Interface (per port) default	Neither a receiver nor a source port
Immediate Leave	Disabled on all ports

IGMP Filtering and Throttling

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a device port can belong. You can control the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a device port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual device ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a device port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing. You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

IGMP filtering controls only group-specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs

the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether CGMP or MVR is used to forward the multicast traffic.

IGMP filtering applies only to the dynamic learning of IP multicast group addresses, not static configuration.

With the IGMP throttling feature, you can set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.



Note IGMPv3 join and leave messages are not supported on devices running IGMP filtering.

Default IGMP Filtering and Throttling Configuration

This table displays the default IGMP filtering and throttling configuration for the device.

Table 22: Default IGMP Filtering Configuration

Feature	Default Setting
IGMP filters	None applied.
IGMP maximum number of IGMP groups	No maximum set. Note When the maximum number of groups is in the forwarding table, the default IGMP throttling action is to deny the report.
IGMP profiles	None defined.
IGMP profile action	Deny the range addresses.

How to Configure IGMP Snooping and MVR

Enabling or Disabling IGMP Snooping on a Device

When IGMP snooping is globally enabled or disabled, it is also enabled or disabled in all existing VLAN interfaces. IGMP snooping is enabled on all VLANs by default, but can be enabled and disabled on a per-VLAN basis.

Global IGMP snooping overrides the VLAN IGMP snooping. If global snooping is disabled, you cannot enable VLAN snooping. If global snooping is enabled, you can enable or disable VLAN snooping.

Follow these steps to globally enable IGMP snooping on the device:

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. `ip igmp snooping`
4. `end`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>ip igmp snooping</code></p> <p>Example:</p> <pre>Switch(config)# ip igmp snooping</pre>	<p>Globally enables IGMP snooping in all existing VLAN interfaces.</p> <p>Note To globally disable IGMP snooping on all VLAN interfaces, use the no ip igmp snooping global configuration command.</p>
Step 4	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 5	<p><code>copy running-config startup-config</code></p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Enabling or Disabling IGMP Snooping on a VLAN Interface

Follow these steps to enable IGMP snooping on a VLAN interface:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip igmp snooping vlan vlan-id`
4. `end`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> Example: <pre>Switch(config)# ip igmp snooping vlan 7</pre>	Enables IGMP snooping on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094. IGMP snooping must be globally enabled before you can enable VLAN snooping. Note To disable IGMP snooping on a VLAN interface, use the no ip igmp snooping vlan <i>vlan-id</i> global configuration command for the specified VLAN number.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting the Snooping Method

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The switch learns of the ports through one of these methods:

- Snooping on IGMP queries, Protocol-Independent Multicast (PIM) packets, and Distance Vector Multicast Routing Protocol (DVMRP) packets.
- Listening to Cisco Group Management Protocol (CGMP) packets from other routers.
- Statically connecting to a multicast router port using the **ip igmp snooping mrouter** global configuration command.

You can configure the switch either to snoop on IGMP queries and PIM/DVMRP packets or to listen to CGMP self-join or proxy-join packets. By default, the switch snoops on PIM/DVMRP packets on all VLANs. To learn of multicast router ports through only CGMP packets, use the **ip igmp snooping vlan *vlan-id* mrouter learn cgmp** global configuration command. When this command is entered, the router listens to only CGMP self-join and CGMP proxy-join packets and to no other CGMP packets. To learn of multicast router ports through only PIM-DVMRP packets, use the **ip igmp snooping vlan *vlan-id* mrouter learn pim-dvmrp** global configuration command.

If you want to use CGMP as the learning method and no multicast routers in the VLAN are CGMP proxy-enabled, you must enter the **ip cgmp router-only** command to dynamically access the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id* mrouter learn {cgmp | pim-dvmrp }**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> mrouter learn {cgmp pim-dvmrp } Example: <pre>Switch(config)# ip igmp snooping vlan 1 mrouter learn cgmp</pre>	Specifies the multicast router learning method: <ul style="list-style-type: none"> • cgmp—Listens for CGMP packets. This method is useful for reducing control traffic. • pim-dvmrp—Snoops on IGMP queries and PIM-DVMRP packets. This is the default. <p>Note To return to the default learning method, use the no ip igmp snooping vlan <i>vlan-id</i> mrouter learn cgmp global configuration command.</p>
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 5	show ip igmp snooping Example: Switch# show ip igmp snooping	Verifies the configuration.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Multicast Router Port

Perform these steps to add a multicast router port (enable a static connection to a multicast router) on the device.



Note Static connections to multicast routers are supported only on device ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id* mrouter interface *interface-id***
4. **end**
5. **show ip igmp snooping mrouter [vlan *vlan-id*]**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	<p>ip igmp snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# ip igmp snooping vlan 5 mrouter interface gigabitethernet1/0/1</pre>	<p>Specifies the multicast router VLAN ID and the interface to the multicast router.</p> <ul style="list-style-type: none"> • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 128. <p>Note To remove a multicast router port from the VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i> global configuration command.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show ip igmp snooping mrouter [<i>vlan vlan-id</i>]</p> <p>Example:</p> <pre>Switch# show ip igmp snooping mrouter vlan 5</pre>	Verifies that IGMP snooping is enabled on the VLAN interface.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Host Statically to Join a Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure a host on an interface.

Follow these steps to add a Layer 2 port as a member of a multicast group:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip igmp snooping vlan vlan-id static ip_address interface interface-id`
4. `end`
5. `show ip igmp snooping groups`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> static <i>ip_address</i> interface <i>interface-id</i> Example: <pre>Switch(config)# ip igmp snooping vlan 105 static 230.0.0.1 interface gigabitethernet1/0/1</pre>	Statically configures a Layer 2 port as a member of a multicast group: <ul style="list-style-type: none"> • <i>vlan-id</i> is the multicast group VLAN ID. The range is 1 to 1001 and 1006 to 4094. • <i>ip-address</i> is the group IP address. • <i>interface-id</i> is the member port. It can be a physical interface or a port channel (1 to 128). <p>Note To remove the Layer 2 port from the multicast group, use the no ip igmp snooping vlan <i>vlan-id</i> static <i>mac-address</i> interface <i>interface-id</i> global configuration command.</p>
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping groups Example: <pre>Switch# show ip igmp snooping groups</pre>	Verifies the member port and the IP address.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling IGMP Immediate Leave

When you enable IGMP Immediate Leave, the device immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should use the Immediate-Leave feature only when there is a single receiver present on every port in the VLAN.



Note Immediate Leave is supported only on IGMP Version 2 hosts. IGMP Version 2 is the default version for the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id* immediate-leave**
4. **end**
5. **show ip igmp snooping vlan *vlan-id***
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> immediate-leave Example: Switch(config)# ip igmp snooping vlan 21 immediate-leave	Enables IGMP Immediate Leave on the VLAN interface. Note To disable IGMP Immediate Leave on a VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> immediate-leave global configuration command.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show ip igmp snooping vlan <i>vlan-id</i> Example: <pre>Switch# show ip igmp snooping vlan 21</pre>	Verifies that Immediate Leave is enabled on the VLAN interface.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the IGMP Leave Timer

You can configure the leave time globally or on a per-VLAN basis. Follow these steps to enable the IGMP configurable-leave timer:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping last-member-query-interval *time***
4. **ip igmp snooping vlan *vlan-id* last-member-query-interval *time***
5. **end**
6. **show ip igmp snooping**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping last-member-query-interval <i>time</i> Example: <pre>Switch(config)# ip igmp snooping</pre>	Configures the IGMP leave timer globally. The range is 100 to 32767 milliseconds. The default leave time is 1000 milliseconds.

	Command or Action	Purpose
	<code>last-member-query-interval 1000</code>	Note To globally reset the IGMP leave timer to the default setting, use the no ip igmp snooping last-member-query-interval global configuration command.
Step 4	<p>ip igmp snooping vlan <i>vlan-id</i> last-member-query-interval <i>time</i></p> <p>Example:</p> <pre>Switch(config)# ip igmp snooping vlan 210 last-member-query-interval 1000</pre>	<p>(Optional) Configures the IGMP leave time on the VLAN interface. The range is 100 to 32767 milliseconds.</p> <p>Note Configuring the leave time on a VLAN overrides the globally configured timer.</p> <p>Note To remove the configured IGMP leave-time setting from the specified VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> global configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show ip igmp snooping</p> <p>Example:</p> <pre>Switch# show ip igmp snooping</pre>	(Optional) Displays the configured IGMP leave time.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring TCN-Related Commands

Controlling the Multicast Flooding Time After a TCN Event

You can configure the number of general queries by which multicast data traffic is flooded after a topology change notification (TCN) event. If you set the TCN flood query count to 1 the flooding stops after receiving 1 general query. If you set the count to 7, the flooding continues until 7 general queries are received. Groups are relearned based on the general queries received during the TCN event.

Some examples of TCN events are when the client location is changed and the receiver is on same port that was blocked but is now forwarding, and when a port goes down without sending a leave message.

Follow these steps to configure the TCN flood query count:

SUMMARY STEPS

1. enable
2. configure terminal
3. ip igmp snooping tcn flood query count *count*
4. end
5. show ip igmp snooping
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping tcn flood query count <i>count</i> Example: <pre>Switch(config)# ip igmp snooping tcn flood query count 3</pre>	Specifies the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. The default, the flooding query count is 2. Note To return to the default flooding query count, use the no ip igmp snooping tcn flood query count global configuration command.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping Example: <pre>Switch# show ip igmp snooping</pre>	Verifies the TCN settings.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Recovering from Flood Mode

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, you can enable the device to send the global leave message whether it is the spanning-tree root or not. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the device is the spanning-tree root regardless of this configuration.

Follow these steps to enable sending of leave messages:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping tcn query solicit**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping tcn query solicit Example: <pre>Switch(config)# ip igmp snooping tcn query solicit</pre>	Sends an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled. Note To return to the default query solicitation, use the no ip igmp snooping tcn query solicit global configuration command.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping Example:	Verifies the TCN settings.

	Command or Action	Purpose
	Switch# <code>show ip igmp snooping</code>	
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Disabling Multicast Flooding During a TCN Event

When the device receives a TCN, multicast traffic is flooded to all the ports until 2 general queries are received. If the device has many ports with attached hosts that are subscribed to different multicast groups, this flooding might exceed the capacity of the link and cause packet loss. Follow these steps to control TCN flooding:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `no ip igmp snooping tcn flood`
5. `end`
6. `show ip igmp snooping`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface interface-id Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the interface to be configured, and enters interface configuration mode.

	Command or Action	Purpose
Step 4	no ip igmp snooping tcn flood Example: <pre>Switch(config-if)# no ip igmp snooping tcn flood</pre>	Disables the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface. Note To re-enable multicast flooding on an interface, use the ip igmp snooping tcn flood interface configuration command.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show ip igmp snooping Example: <pre>Switch# show ip igmp snooping</pre>	Verifies the TCN settings.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Snooping Querier

Follow these steps to enable the IGMP snooping querier feature in a VLAN:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping querier**
4. **ip igmp snooping querier address** *ip_address*
5. **ip igmp snooping querier query-interval** *interval-count*
6. **ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*]
7. **ip igmp snooping querier timer expiry** *timeout*
8. **ip igmp snooping querier version** *version*
9. **end**
10. **show ip igmp snooping vlan** *vlan-id*
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping querier Example: Switch(config)# ip igmp snooping querier	Enables the IGMP snooping querier.
Step 4	ip igmp snooping querier address <i>ip_address</i> Example: Switch(config)# ip igmp snooping querier address 172.16.24.1	(Optional) Specifies an IP address for the IGMP snooping querier. If you do not specify an IP address, the querier tries to use the global IP address configured for the IGMP querier. Note The IGMP snooping querier does not generate an IGMP general query if it cannot find an IP address on the device.
Step 5	ip igmp snooping querier query-interval <i>interval-count</i> Example: Switch(config)# ip igmp snooping querier query-interval 30	(Optional) Sets the interval between IGMP queriers. The range is 1 to 18000 seconds.
Step 6	ip igmp snooping querier tcn query [count <i>count</i> interval <i>interval</i>] Example: Switch(config)# ip igmp snooping querier tcn query interval 20	(Optional) Sets the time between Topology Change Notification (TCN) queries. The count range is 1 to 10. The interval range is 1 to 255 seconds.
Step 7	ip igmp snooping querier timer expiry <i>timeout</i> Example: Switch(config)# ip igmp snooping querier timer expiry 180	(Optional) Sets the length of time until the IGMP querier expires. The range is 60 to 300 seconds.

	Command or Action	Purpose
Step 8	ip igmp snooping querier version <i>version</i> Example: <pre>Switch(config)# ip igmp snooping querier version 2</pre>	(Optional) Selects the IGMP version number that the querier feature uses. Select 1 or 2.
Step 9	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	show ip igmp snooping vlan <i>vlan-id</i> Example: <pre>Switch# show ip igmp snooping vlan 30</pre>	(Optional) Verifies that the IGMP snooping querier is enabled on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094.
Step 11	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Disabling IGMP Report Suppression

Follow these steps to disable IGMP report suppression:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no ip igmp snooping report-suppression**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no ip igmp snooping report-suppression Example: Switch(config)# <code>no ip igmp snooping report-suppression</code>	Disables IGMP report suppression. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers. IGMP report suppression is enabled by default. When IGMP report suppression is enabled, the device forwards only one IGMP report per multicast router query. Note To re-enable IGMP report suppression, use the ip igmp snooping report-suppression global configuration command.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping Example: Switch# <code>show ip igmp snooping</code>	Verifies that IGMP report suppression is disabled.
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring MVR Global Parameters

You do not need to set the optional MVR parameters if you choose to use the default settings. If you want to change the default parameters (except for the MVR VLAN), you must first enable MVR.



Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mvr**
4. **mvr group** *ip-address* [*count*]
5. **mvr querytime** *value*
6. **mvr vlan** *vlan-id*
7. **mvr mode** {*dynamic* | *compatible*}
8. **end**
9. Use one of the following:
 - **show mvr**
 - **show mvr members**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	mvr Example: Switch (config)# mvr	Enables MVR on the device.
Step 4	mvr group <i>ip-address</i> [<i>count</i>] Example: Switch(config)# mvr group 228.1.23.4	Configures an IP multicast address on the device or use the <i>count</i> parameter to configure a contiguous series of MVR group addresses (the range for <i>count</i> is 1 to 256; the default is 1). Any multicast data sent to this address is sent to all source ports on the device and all receiver ports that have elected to receive data on that multicast address. Each multicast address would correspond to one television channel. Note To return the switch to its default settings, use the no mvr [<i>mode</i> <i>group ip-address</i> <i>querytime</i> <i>vlan</i>] global configuration commands.

	Command or Action	Purpose
Step 5	mvr querytime <i>value</i> Example: <pre>Switch(config)# mvr querytime 10</pre>	(Optional) Defines the maximum time to wait for IGMP report memberships on a receiver port before removing the port from multicast group membership. The value is in units of tenths of a second. The range is 1 to 100, and the default is 5 tenths or one-half second.
Step 6	mvr vlan <i>vlan-id</i> Example: <pre>Switch(config)# mvr vlan 22</pre>	(Optional) Specifies the VLAN in which multicast data is received; all source ports must belong to this VLAN. The VLAN range is 1 to 1001 and 1006 to 4094. The default is VLAN 1.
Step 7	mvr mode { dynamic compatible } Example: <pre>Switch(config)# mvr mode dynamic</pre>	(Optional) Specifies the MVR mode of operation: <ul style="list-style-type: none"> • dynamic—Allows dynamic MVR membership on source ports. • compatible—Is compatible with Catalyst 3500 XL and Catalyst 2900 XL devices and does not support IGMP dynamic joins on source ports. The default is compatible mode. Note To return the switch to its default settings, use the no mvr [mode group ip-address querytime vlan] global configuration commands.
Step 8	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	Use one of the following: <ul style="list-style-type: none"> • show mvr • show mvr members Example: <pre>Switch# show mvr</pre> OR <pre>Switch# show mvr members</pre>	Verifies the configuration.
Step 10	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Configuring MVR Interfaces

Follow these steps to configure Layer 2 MVR interfaces:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `mvr`
4. `interface interface-id`
5. `mvr type {source | receiver}`
6. `mvr vlan vlan-id group [ip-address]`
7. `mvr immediate`
8. `end`
9. Use one of the following:
 - `show mvr`
 - `show mvr interface`
 - `show mvr members`
10. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	mvr Example: Switch (config)# <code>mvr</code>	Enables MVR on the device.

	Command or Action	Purpose
Step 4	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet1/0/2</pre>	Specifies the Layer 2 port to configure, and enter interface configuration mode.
Step 5	<p>mvr type {source receiver}</p> <p>Example:</p> <pre>Switch(config-if)# mvr type receiver</pre>	<p>Configures an MVR port as one of these:</p> <ul style="list-style-type: none"> • source—Configures uplink ports that receive and send multicast data as source ports. Subscribers cannot be directly connected to source ports. All source ports on a device belong to the single multicast VLAN. • receiver—Configures a port as a receiver port if it is a subscriber port and should only receive multicast data. It does not receive data unless it becomes a member of the multicast group, either statically or by using IGMP leave and join messages. Receiver ports cannot belong to the multicast VLAN. <p>The default configuration is as a non-MVR port. If you attempt to configure a non-MVR port with MVR characteristics, the operation fails.</p> <p>Note To return the interface to its default settings, use the no mvr [type immediate vlan <i>vlan-id</i> group] interface configuration commands.</p>
Step 6	<p>mvr vlan <i>vlan-id</i> group [<i>ip-address</i>]</p> <p>Example:</p> <pre>Switch(config-if)# mvr vlan 22 group 228.1.23.4</pre>	<p>(Optional) Statically configures a port to receive multicast traffic sent to the multicast VLAN and the IP multicast address. A port statically configured as a member of a group remains a member of the group until statically removed.</p> <p>Note In compatible mode, this command applies to only receiver ports. In dynamic mode, it applies to receiver ports and source ports.</p> <p>Receiver ports can also dynamically join multicast groups by using IGMP join and leave messages.</p>
Step 7	<p>mvr immediate</p> <p>Example:</p> <pre>Switch(config-if)# mvr immediate</pre>	<p>(Optional) Enables the Immediate-Leave feature of MVR on the port.</p> <p>Note This command applies to only receiver ports and should only be enabled on receiver ports to which a single receiver device is connected.</p>

	Command or Action	Purpose
Step 8	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	Use one of the following: <ul style="list-style-type: none"> • show mvr • show mvr interface • show mvr members Example: <pre>Switch# show mvr interface Port Type Status Immediate Leave ----- ----- Gi1/0/2 RECEIVER ACTIVE/DOWN ENABLED</pre>	Verifies the configuration.
Step 10	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring IGMP Profiles

Follow these steps to create an IGMP profile:

This task is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp profile** *profile number*
4. **permit | deny**
5. **range** *ip multicast address*
6. **end**
7. **show ip igmp profile** *profile number*
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip igmp profile <i>profile number</i> Example: Switch(config)# ip igmp profile 3	Assigns a number to the profile you are configuring, and enters IGMP profile configuration mode. The profile number range is 1 to 4294967295. When you are in IGMP profile configuration mode, you can create the profile by using these commands: <ul style="list-style-type: none"> • deny—Specifies that matching addresses are denied; this is the default. • exit—Exits from igmp-profile configuration mode. • no—Negates a command or returns to its defaults. • permit—Specifies that matching addresses are permitted. • range—Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address. The default is for the device to have no IGMP profiles configured. Note To delete a profile, use the no ip igmp profile <i>profile number</i> global configuration command.
Step 4	permit deny Example: Switch(config-igmp-profile)# permit	(Optional) Sets the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.
Step 5	range <i>ip multicast address</i> Example: Switch(config-igmp-profile)# range 229.9.9.0	Enters the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address.

	Command or Action	Purpose
		<p>You can use the range command multiple times to enter multiple addresses or ranges of addresses.</p> <p>Note To delete an IP multicast address or range of IP multicast addresses, use the no range ip multicast address IGMP profile configuration command.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show ip igmp profile <i>profile number</i></p> <p>Example:</p> <pre>Switch# show ip igmp profile 3</pre>	Verifies the profile configuration.
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 9	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Applying IGMP Profiles

To control access as defined in an IGMP profile, you have to apply the profile to the appropriate interfaces. You can apply IGMP profiles only to Layer 2 access ports; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can have only one profile applied to it.

Follow these steps to apply an IGMP profile to a switch port:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **ip igmp filter *profile number***
5. **end**

6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet1/0/1</code>	Specifies the physical interface, and enters interface configuration mode. The interface must be a Layer 2 port that does not belong to an EtherChannel port group.
Step 4	ip igmp filter <i>profile number</i> Example: Switch(config-if)# <code>ip igmp filter 321</code>	Applies the specified IGMP profile to the interface. The range is 1 to 4294967295. Note To remove a profile from an interface, use the no ip igmp filter <i>profile number</i> interface configuration command.
Step 5	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Setting the Maximum Number of IGMP Groups

Follow these steps to set the maximum number of IGMP groups that a Layer 2 interface can join:

Before you begin

This restriction can be applied to Layer 2 ports only; you cannot set a maximum number of IGMP groups on routed ports or SVIs. You also can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp max-groups** *number*
5. **end**
6. **show running-config interface** *interface-id*
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/2</pre>	Specifies the interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or a EtherChannel interface.
Step 4	ip igmp max-groups <i>number</i> Example: <pre>Switch(config-if)# ip igmp max-groups 20</pre>	Sets the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set. Note To remove the maximum group limitation and return to the default of no maximum, use the no ip igmp max-groups interface configuration command.

	Command or Action	Purpose
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config interface <i>interface-id</i> Example: Switch# interface gigabitethernet1/0/1	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received.

Follow these steps to configure the throttling action when the maximum number of entries is in the forwarding table:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **ip igmp max-groups action {deny | replace}**
5. **end**
6. **show running-config interface *interface-id***
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet1/0/1</code>	Specifies the physical interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or an EtherChannel interface. The interface cannot be a trunk port.
Step 4	ip igmp max-groups action {deny replace} Example: Switch(config-if)# <code>ip igmp max-groups action replace</code>	<p>When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specifies the action that the interface takes:</p> <ul style="list-style-type: none"> • deny—Drops the report. If you configure this throttling action, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the device drops the next IGMP report received on the interface. • replace—Replaces the existing group with the new group for which the IGMP report was received. If you configure this throttling action, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the device replaces a randomly selected entry with the received IGMP report. <p>To prevent the device from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table.</p> <p>Note To return to the default action of dropping the report, use the no ip igmp max-groups action interface configuration command.</p>
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config interface <i>interface-id</i> Example: Switch# <code>show running-config interface gigabitethernet1/0/1</code>	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring IGMP Snooping and MVR

Monitoring IGMP Snooping Information

You can display IGMP snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display MAC address multicast entries for a VLAN configured for IGMP snooping.

Table 23: Commands for Displaying IGMP Snooping Information

Command	Purpose
show ip igmp snooping [vlan <i>vlan-id</i> [detail]]	Displays the snooping configuration information for all VLANs on the device or for a specified VLAN. (Optional) Enter vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ip igmp snooping groups [count dynamic [count] user [count]]	Displays multicast table information for the device or about a specific parameter: <ul style="list-style-type: none"> • count—Displays the total number of entries for the specified command options instead of the actual entries. • dynamic—Displays entries learned through IGMP snooping. • user—Displays only the user-configured multicast entries.

Command	Purpose
show ip igmp snooping groups <i>vlan</i> <i>vlan-id</i> [<i>ip_address</i> count dynamic [count] user [count]]	<p>Displays multicast table information for a multicast VLAN or about a specific parameter for the VLAN:</p> <ul style="list-style-type: none"> • <i>vlan-id</i>—The VLAN ID range is 1 to 1001 and 1006 to 4094. • count—Displays the total number of entries for the specified command options instead of the actual entries. • dynamic—Displays entries learned through IGMP snooping. • <i>ip_address</i>—Displays characteristics of the multicast group with the specified group IP address. • user—Displays only the user-configured multicast entries.
show ip igmp snooping mrouter [<i>vlan</i> <i>vlan-id</i>]	<p>Displays information on dynamically learned and manually configured multicast router interfaces.</p> <p>Note When you enable IGMP snooping, the device automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces.</p> <p>(Optional) Enter the vlan <i>vlan-id</i> to display information for a particular VLAN.</p>
show ip igmp snooping querier [<i>vlan</i> <i>vlan-id</i>] detail	<p>Displays information about the IP address and receiving port of the most-recently received IGMP query message in the VLAN and the configuration and operational state of the IGMP snooping querier in the VLAN.</p>

Monitoring MVR

You can monitor MVR for the switch or for a specified interface by displaying the following MVR information.

Table 24: Commands for Displaying MVR Information

Command	Purpose
show mvr	<p>Displays MVR status and values for the switch—whether MVR is enabled for a multicast VLAN, the maximum (256) and current (0 through 256) number of multicast entries, the query response time, and the MVR mode.</p>

Command	Purpose
show mvr interface [<i>interface-id</i>] [members [<i>vlan vlan-id</i>]]	<p>Displays all MVR interfaces and their MVR configurations.</p> <p>When a specific interface is entered, displays this information:</p> <ul style="list-style-type: none"> • Type—Receiver or Source • Status—One of these: <ul style="list-style-type: none"> • Active means the port is part of a VLAN. • Up/Down means that the port is forwarding or nonforwarding. • Inactive means that the port is not part of any VLAN. • Immediate Leave—Enabled or Disabled <p>If the members keyword is entered, displays all multicast group members. If a VLAN identification is entered, all multicast group members of that VLAN. The range is 1 to 1001 and 1006 to 4094.</p>
show mvr members [<i>ip-address</i>]	Displays all receiver and source ports that are members of any IP multicast group IP address.

Monitoring IGMP Filtering and Throttling Configuration

You can display IGMP profile characteristics, and you can display the IGMP profile and maximum group configuration for all interfaces on the device or for a specified interface. You can also display the IGMP throttling configuration for all interfaces on the device or for a specified interface.

Table 25: Commands for Displaying IGMP Filtering and Throttling Configuration

Command	Purpose
show ip igmp profile [<i>profile number</i>]	Displays the specified IGMP profile or all the IGMP profiles defined on the device.
show running-config [interface <i>interface-id</i>]	Displays the configuration of the specified interface. If no interface is specified, displays the configuration of all interfaces on the device, including (if configured) the maximum number of IGMP group members that an interface can belong and the IGMP profile applied to the interface.

Configuration Examples for IGMP Snooping and MVR

Example: Configuring IGMP Snooping Using CGMP Packets

This example shows how to configure IGMP snooping to use CGMP packets as the learning method:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 1 mrouter learn cgmp
Switch(config)# end
```

Example: Enabling a Static Connection to a Multicast Router

This example shows how to enable a static connection to a multicast router:

```
Switch configure terminal
Switch ip igmp snooping vlan 200 mrouter interface gigabitethernet1/0/2
Switch end
```

Example: Configuring a Host Statically to Join a Group

This example shows how to statically configure a host on a port:

```
Switch# configure terminal
Switch# ip igmp snooping vlan 105 static 0100.1212.0000 interface gigabitethernet1/0/1
Switch# end
```

Example: Enabling IGMP Immediate Leave

This example shows how to enable IGMP Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 130 immediate-leave
Switch(config)# end
```

Example: Setting the IGMP Snooping Querier Source Address

This example shows how to set the IGMP snooping querier source address to 10.0.0.64:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier 10.0.0.64
Switch(config)# end
```

Example: Setting the IGMP Snooping Querier Maximum Response Time

This example shows how to set the IGMP snooping querier maximum response time to 25 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier query-interval 25
Switch(config)# end
```

Example: Setting the IGMP Snooping Querier Timeout

This example shows how to set the IGMP snooping querier timeout to 60 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier timeout expiry 60
Switch(config)# end
```

Example: Setting the IGMP Snooping Querier Feature

This example shows how to set the IGMP snooping querier feature to Version 2:

```
Switch# configure terminal
Switch(config)# no ip igmp snooping querier version 2
Switch(config)# end
```

Example: Configuring IGMP Profiles

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Switch(config)# ip igmp profile 4
Switch(config-igmp-profile)# permit
Switch(config-igmp-profile)# range 229.9.9.0
Switch(config-igmp-profile)# end
Switch# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

Example: Applying IGMP Profile

This example shows how to apply IGMP profile 4 to a port:

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip igmp filter 4
Switch(config-if)# end
```

Example: Setting the Maximum Number of IGMP Groups

This example shows how to limit to 25 the number of IGMP groups that a port can join:

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip igmp max-groups 25
Switch(config-if)# end
```

Example: Configuring MVR Global Parameters

This example shows how to enable MVR, configure the group address, set the query time to 1 second (10 tenths), specify the MVR multicast VLAN as VLAN 22, and set the MVR mode as dynamic:

```
Switch(config)# mvr
Switch(config)# mvr group 228.1.23.4
Switch(config)# mvr querytime 10
Switch(config)# mvr vlan 22
Switch(config)# mvr mode dynamic
Switch(config)# end
```


Example: Configuring MVR Interfaces

This example shows how to configure a port as a receiver port, statically configure the port to receive multicast traffic sent to the multicast group address, configure Immediate Leave on the port, and verify the results:

```
Switch(config)# mvr
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# mvr type receiver
Switch(config-if)# mvr vlan 22 group 228.1.23.4
Switch(config-if)# mvr immediate
Switch(config)# end
Switch# show mvr interface
```

```
Port Type Status Immediate Leave
---- ---- -
Gi1/0/2 RECEIVER ACTIVE/DOWN ENABLED
```




CHAPTER 14

Configuring CGMP

- [Finding Feature Information](#), on page 163
- [Prerequisites for Configuring CGMP](#), on page 163
- [Restrictions for CGMP](#), on page 163
- [Information About CGMP](#), on page 164
- [Enabling CGMP Server Support](#), on page 164
- [Monitoring CGMP](#), on page 166

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Configuring CGMP

The following are the prerequisites for configuring CGMP:

- When multiple Cisco CGMP-capable devices are connected to a switched network and the **ip cgmp proxy** command is needed, we recommend that all devices be configured with the same CGMP option and have precedence for becoming the IGMP querier over non-Cisco routers.
- To use CGMP, you must have IP Services feature set enabled on the 3560-CX switch.

Restrictions for CGMP

The following are the restrictions for CGMP:

- CGMP is mutually exclusive with HSRPv1. You cannot enable CGMP leaving processing and HSRPv1 at the same time. However, you can enable CGMP and HSRPv2 at the same time.

Information About CGMP

Cisco Group Management Protocol or CGMP-server support is provided on the device; no client-side functionality is provided. The device serves as a CGMP server for devices that do not support IGMP snooping but have CGMP-client functionality.

CGMP is a protocol used on Cisco routers and multilayer devices connected to Layer 2 Catalyst devices to perform tasks similar to those performed by IGMP. CGMP permits Layer 2 group membership information to be communicated from the CGMP server to the device. The device can then learn on which interfaces multicast members reside instead of flooding multicast traffic to all device interfaces. (IGMP snooping is another method to constrain the flooding of multicast packets.)

CGMP is necessary because the Layer 2 device cannot distinguish between IP multicast data packets and IGMP report messages, which are both at the MAC level and are addressed to the same group address.

Enabling CGMP Server Support

When multiple Cisco CGMP-capable devices are connected to a switched network and you configure the **ip cgmp proxy** command, we recommend that all devices be configured with the same CGMP option and have precedence for becoming the IGMP querier over non-Cisco routers. Perform these steps to enable the CGMP server on the device interface:

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **ip cgmp [proxy | router-only]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface that is connected to the Layer 2 Catalyst device, and enters interface configuration mode.
Step 4	<p>ip cgmp [proxy router-only]</p> <p>Example:</p> <pre>Switch(config-if)# ip cgmp proxy</pre>	<p>Enables CGMP on the interface.</p> <p>By default, CGMP is disabled on all interfaces.</p> <p>Enabling CGMP triggers a CGMP join message. Enable CGMP only on Layer 3 interfaces connected to Layer 2 Catalyst devices.</p> <p>(Optional) When you enter the proxy keyword, the CGMP proxy function is enabled. The proxy router advertises the existence of non-CGMP-capable routers by sending a CGMP join message with the non-CGMP-capable router MAC address and a group address of 0000.0000.0000.</p> <p>Note To perform CGMP proxy, the device must be the IGMP querier. If you configure the ip cgmp proxy command, you must manipulate the IP addresses so that the device is the IGMP querier, which might be the highest or lowest IP address, depending on which version of IGMP is running on the network. An IGMP Version 2 querier is selected based on the lowest IP address on the interface. An IGMP Version 1 querier is selected based on the multicast routing protocol used on the interface.</p> <p>Note To disable CGMP on the interface, use the no ip cgmp interface configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

What to do next

Verify the Layer 2 Catalyst device CGMP-client configuration. For more information, see the documentation that shipped with the product

Monitoring CGMP

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 26: Commands for Displaying System and Network Statistics

Command	Purpose
<code>ping [group-name group-address]</code>	Sends an ICMP Echo Request to a multicast group address.
<code>show ip igmp groups [group-name group-address type number]</code>	Displays the multicast groups that are directly connected to the switch that were learned through IGMP.
<code>show ip igmp interface [type number]</code>	Displays multicast-related information about an interface.
<code>show ip mcache [group [source]]</code>	Displays the contents of the IP fast-switching cache.
<code>show ip mpacket [source-address name] [group-address name] [detail]</code>	Displays the contents of the circular cache-header buffer.
<code>show ip mroute [group-name group-address] [source] [summary] [count] [active kbps]</code>	Displays the contents of the IP multicast routing table.
<code>show ip pim interface [type number] [count] [detail]</code>	Displays information about interfaces configured for PIM. This command is available in all software images.
<code>show ip pim neighbor [type number]</code>	Lists the PIM neighbors discovered by the switch. This command is available in all software images.
<code>show ip pim rp [group-name group-address]</code>	Displays the RP routers associated with a sparse-mode multicast group. This command is available in all software images.

Command	Purpose
show ip rpf { <i>source-address</i> <i>name</i> }	Displays how the switch is doing Reverse-Path Forwarding (the unicast routing table, DVMRP routing table, or static mroutes)
show ip sap [<i>group</i> <i>session-name</i> detail]	Displays the Session Announcement Protocol (SAP) Version 2



CHAPTER 15

Configuring Protocol Independent Multicast (PIM)

- [Prerequisites for PIM, on page 169](#)
- [Restrictions for PIM, on page 170](#)
- [Information About PIM, on page 172](#)
- [How to Configure PIM, on page 185](#)
- [Monitoring and Troubleshooting PIM, on page 217](#)
- [Configuration Examples for PIM, on page 218](#)

Prerequisites for PIM

- Before you begin the PIM configuration process, decide which PIM mode to use. This is based on the applications you intend to support on your network. Use the following guidelines:
 - In general, if the application is one-to-many or many-to-many in nature, then PIM-SM can be used successfully.
 - For optimal one-to-many application performance, SSM is appropriate but requires IGMP version 3 support.
- Before you configure PIM stub routing, check that you have met these conditions:
 - You must have IP multicast routing configured on both the stub router and the central router. You must also have PIM mode (dense-mode, sparse-mode, or sparse-dense-mode) configured on the uplink interface of the stub router.
 - You must also configure Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing on the device.
 - The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior.

Restrictions for PIM

PIMv1 and PIMv2 Interoperability

To avoid misconfiguring multicast routing on your device, review the information in this section.

The Cisco PIMv2 implementation provides interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. PIM Versions 1 and 2 can be configured on different routers and multilayer devices within one network. Internally, all routers and multilayer devices on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer devices in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF.



Note We recommend that you use PIMv2. The BSR function interoperates with Auto-RP on Cisco routers and multilayer devices.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer device in the group. Not all routers and devices in the domain use the PIMv2 hash function to select multiple RPs.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we recommend:

- Using Auto-RP throughout the region.
- Configuring sparse-dense mode throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP.

Restrictions for Configuring PIM Stub Routing

- The IP services image contains complete multicast routing.
- Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM protocol is not supported in access domains.
- In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a device that is configured with PIM stub routing.
- The redundant PIM stub router topology is not supported. Only the nonredundant access router topology is supported by the PIM stub feature.

Restrictions for Configuring Auto-RP and BSR

Take into consideration your network configuration, and the following restrictions when configuring Auto-RP and BSR:

Restrictions for Configuring Auto-RP

The following are restrictions for configuring Auto-RP (if used in your network configuration):

- If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must manually configure an RP.
- If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.
- If routed interfaces are configured in sparse mode and you enter the **ip pim autorp listener** global configuration command, Auto-RP can still be used even if all devices are not configured with a manual RP address for the Auto-RP groups.

Restrictions for Configuring BSR

The following are the restrictions for configuring BSR (if used in your network configuration):

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP.
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, have backup RPs serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

Restrictions and Guidelines for Configuring Auto-RP and BSR

The following are restrictions for configuring Auto-RP and BSR (if used in your network configuration):

- If your network is all Cisco routers and multilayer devices, you can use either Auto-RP or BSR.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers and multilayer devices and non-Cisco routers, you must use both Auto-RP and BSR. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure that no PIMv1 device is located in the path a between the BSR and a non-Cisco PIMv2 device.



Note There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents these messages from reaching all routers and multilayer devices in your network. Therefore, if your network has a PIMv1 device in it and only Cisco routers and multilayer devices, it is best to use Auto-RP.

- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer device. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.
- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer devices, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR.

Information About PIM

Protocol Independent Multicast

The Protocol Independent Multicast (PIM) protocol maintains the current IP multicast service mode of receiver-initiated membership. PIM is not dependent on a specific unicast routing protocol; it is IP routing protocol independent and can leverage whichever unicast routing protocols are used to populate the unicast routing table, including Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), Border Gateway Protocol (BGP), and static routes. PIM uses unicast routing information to perform the multicast forwarding function.

Although PIM is called a multicast routing protocol, it actually uses the unicast routing table to perform the reverse path forwarding (RPF) check function instead of building up a completely independent multicast routing table. Unlike other routing protocols, PIM does not send and receive routing updates between routers.

PIM can operate in dense mode or sparse mode. The router can also handle both sparse groups and dense groups at the same time. The mode determines how the router populates its multicast routing table and how the router forwards multicast packets it receives from its directly connected LANs.

PIM is supported only on 3560-CX switches.

For information about PIM forwarding (interface) modes, see the following sections:

PIM Dense Mode

PIM dense mode (PIM-DM) uses a push model to flood multicast traffic to every corner of the network. This push model is a method for delivering data to the receivers without the receivers requesting the data. This method is efficient in certain deployments in which there are active receivers on every subnet in the network.

In dense mode, a router assumes that all other routers want to forward multicast packets for a group. If a router receives a multicast packet and has no directly connected members or PIM neighbors present, a prune message is sent back to the source. Subsequent multicast packets are not flooded to this router on this pruned branch. PIM builds source-based multicast distribution trees.

PIM-DM initially floods multicast traffic throughout the network. Routers that have no downstream neighbors prune back the unwanted traffic. This process repeats every 3 minutes.

Routers accumulate state information by receiving data streams through the flood and prune mechanism. These data streams contain the source and group information so that downstream routers can build up their multicast forwarding table. PIM-DM supports only source trees--that is, (S,G) entries--and cannot be used to build a shared distribution tree.



Note Dense mode is not often used and its use is not recommended. For this reason it is not specified in the configuration tasks in related modules.

PIM Sparse Mode

PIM sparse mode (PIM-SM) uses a pull model to deliver multicast traffic. Only network segments with active receivers that have explicitly requested the data will receive the traffic.

Sparse mode interfaces are added to the multicast routing table only when periodic Join messages are received from downstream routers, or when a directly connected member is on the interface. When forwarding from a LAN, sparse mode operation occurs if an RP is known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense mode fashion. If the multicast traffic from a specific source is sufficient, the first hop router of the receiver may send Join messages toward the source to build a source-based distribution tree.

PIM-SM distributes information about active sources by forwarding data packets on the shared tree. Because PIM-SM uses shared trees (at least, initially), it requires the use of a rendezvous point (RP). The RP must be administratively configured in the network. See the [Rendezvous Points, on page 176](#) section for more information.

In sparse mode, a router assumes that other routers do not want to forward multicast packets for a group, unless there is an explicit request for the traffic. When hosts join a multicast group, the directly connected routers send PIM Join messages toward the RP. The RP keeps track of multicast groups. Hosts that send multicast packets are registered with the RP by the first hop router of that host. The RP then sends Join messages toward the source. At this point, packets are forwarded on a shared distribution tree. If the multicast traffic from a specific source is sufficient, the first hop router of the host may send Join messages toward the source to build a source-based distribution tree.

Sources register with the RP and then data is forwarded down the shared tree to the receivers. The edge routers learn about a particular source when they receive data packets on the shared tree from that source through the RP. The edge router then sends PIM (S,G) Join messages toward that source. Each router along the reverse path compares the unicast routing metric of the RP address to the metric of the source address. If the metric for the source address is better, it will forward a PIM (S,G) Join message toward the source. If the metric for the RP is the same or better, then the PIM (S,G) Join message will be sent in the same direction as the RP. In this case, the shared tree and the source tree would be considered congruent.

If the shared tree is not an optimal path between the source and the receiver, the routers dynamically create a source tree and stop traffic from flowing down the shared tree. This behavior is the default behavior in software. Network administrators can force traffic to stay on the shared tree by using the **ip pim spt-threshold infinity** command.

PIM-SM scales well to a network of any size, including those with WAN links. The explicit join mechanism prevents unwanted traffic from flooding the WAN links.

Sparse-Dense Mode

If you configure either sparse mode or dense mode on an interface, then sparseness or denseness is applied to the interface as a whole. However, some environments might require PIM to run in a single region in sparse mode for some groups and in dense mode for other groups.

An alternative to enabling only dense mode or only sparse mode is to enable sparse-dense mode. In this case, the interface is treated as dense mode if the group is in dense mode; the interface is treated in sparse mode if

the group is in sparse mode. You must have an RP if the interface is in sparse-dense mode and you want to treat the group as a sparse group.

If you configure sparse-dense mode, the idea of sparseness or denseness is applied to the groups for which the router is a member.

Another benefit of sparse-dense mode is that Auto-RP information can be distributed in a dense mode; yet, multicast groups for user groups can be used in a sparse mode manner. Therefore there is no need to configure a default RP at the leaf routers.

When an interface is treated in dense mode, it is populated in the outgoing interface list of a multicast routing table when either of the following conditions is true:

- Members or DVMRP neighbors are on the interface.
- There are PIM neighbors and the group has not been pruned.

When an interface is treated in sparse mode, it is populated in the outgoing interface list of a multicast routing table when either of the following conditions is true:

- Members or DVMRP neighbors are on the interface.
- An explicit Join message has been received by a PIM neighbor on the interface.

PIM Versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution function that enables routers and multilayer devices to dynamically learn the group-to-RP mappings.
- Sparse mode and dense mode are properties of a group, as opposed to an interface.



Note We strongly recommend using sparse-dense mode as opposed to either sparse mode or dense mode only.

- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages sent to an RP specify whether they are sent by a border router or a designated router.
- PIM packets are no longer inside IGMP packets; they are standalone packets.

PIM Stub Routing

The PIM stub routing feature, available in all of the device software images, reduces resource usage by moving routed traffic closer to the end user.

The PIM stub routing feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces, uplink PIM interfaces, and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic, it only passes and forwards IGMP traffic.

In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a device that is configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

When using PIM stub routing, you should configure the distribution and remote routers to use IP multicast routing and configure only the device as a PIM stub router. The device does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the device. The device uplink port cannot be used with SVIs. If you need PIM for an SVI uplink port, you should upgrade to the IP Services feature set.

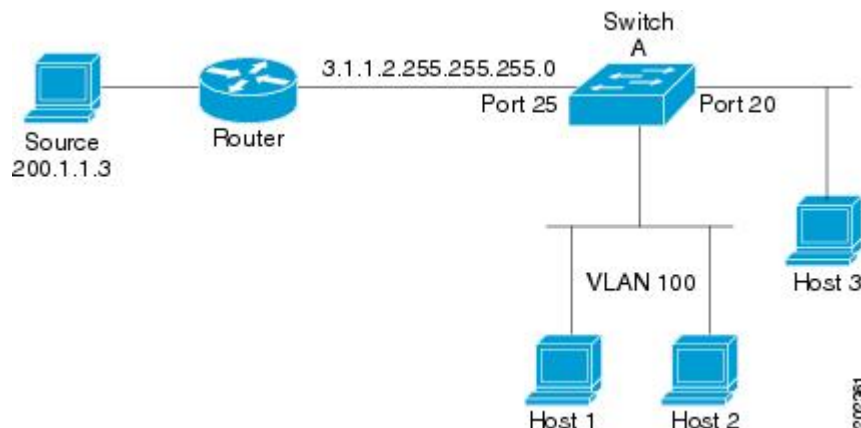


Note You must also configure EIGRP stub routing when configuring PIM stub routing on the device

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM asset and designated router election mechanisms are not supported on the PIM passive interfaces. Only the nonredundant access router topology is supported by the PIM stub feature. By using a nonredundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

Figure 6: PIM Stub Router Configuration

In the following figure, the Device A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source 200.1.1.3.



IGMP Helper

PIM stub routing moves routed traffic closer to the end user and reduces network traffic. You can also reduce traffic by configuring a stub router (switch) with the IGMP helper feature.

You can configure a stub router (switch) with the **ip igmp helper-address** *ip-address* interface configuration command to enable the switch to send reports to the next-hop interface. Hosts that are not directly connected to a downstream router can then join a multicast group sourced from an upstream network. The IGMP packets from a host wanting to join a multicast stream are forwarded upstream to the next-hop device when this feature is configured. When the upstream central router receives the helper IGMP reports or leaves, it adds or removes the interfaces from its outgoing interface list for that group.

Rendezvous Points

A rendezvous point (RP) is a role that a device performs when operating in Protocol Independent Multicast (PIM) Sparse Mode (SM). An RP is required only in networks running PIM SM. In the PIM-SM model, only network segments with active receivers that have explicitly requested multicast data will be forwarded the traffic.

This method of delivering multicast data is in contrast to PIM Dense Mode (PIM DM). In PIM DM, multicast traffic is initially flooded to all segments of the network. Routers that have no downstream neighbors or directly connected receivers prune back the unwanted traffic.

An RP acts as the meeting place for sources and receivers of multicast data. In a PIM-SM network, sources must send their traffic to the RP. This traffic is then forwarded to receivers down a shared distribution tree. By default, when the first hop device of the receiver learns about the source, it will send a Join message directly to the source, creating a source-based distribution tree from the source to the receiver. This source tree does not include the RP unless the RP is located within the shortest path between the source and receiver.

In most cases, the placement of the RP in the network is not a complex decision. By default, the RP is needed only to start new sessions with sources and receivers. Consequently, the RP experiences little overhead from traffic flow or processing. In PIM version 2, the RP performs less processing than in PIM version 1 because sources must only periodically register with the RP to create state.

Auto-RP

In the first version of PIM-SM, all leaf routers (routers directly connected to sources or receivers) were required to be manually configured with the IP address of the RP. This type of configuration is also known as static RP configuration. Configuring static RPs is relatively easy in a small network, but it can be laborious in a large, complex network.

Following the introduction of PIM-SM version 1, Cisco implemented a version of PIM-SM with the Auto-RP feature. Auto-RP automates the distribution of group-to-RP mappings in a PIM network. Auto-RP has the following benefits:

- Configuring the use of multiple RPs within a network to serve different groups is easy.
- Auto-RP allows load splitting among different RPs and arrangement of RPs according to the location of group participants.
- Auto-RP avoids inconsistent, manual RP configurations that can cause connectivity problems.

Multiple RPs can be used to serve different group ranges or serve as backups to each other. For Auto-RP to work, a router must be designated as an RP-mapping agent, which receives the RP-announcement messages from the RPs and arbitrates conflicts. The RP-mapping agent then sends the consistent group-to-RP mappings to all other routers. Thus, all routers automatically discover which RP to use for the groups they support.



Note If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must statically configure an RP.



Note If router interfaces are configured in sparse mode, Auto-RP can still be used if all routers are configured with a static RP address for the Auto-RP groups.

To make Auto-RP work, a router must be designated as an RP mapping agent, which receives the RP announcement messages from the RPs and arbitrates conflicts. The RP mapping agent then sends the consistent group-to-RP mappings to all other routers by dense mode flooding. Thus, all routers automatically discover which RP to use for the groups they support. The Internet Assigned Numbers Authority (IANA) has assigned two group addresses, 224.0.1.39 and 224.0.1.40, for Auto-RP. One advantage of Auto-RP is that any change to the RP designation must be configured only on the routers that are RPs and not on the leaf routers. Another advantage of Auto-RP is that it offers the ability to scope the RP address within a domain. Scoping can be achieved by defining the time-to-live (TTL) value allowed for the Auto-RP advertisements.

Each method for configuring an RP has its own strengths, weaknesses, and level of complexity. In conventional IP multicast network scenarios, we recommend using Auto-RP to configure RPs because it is easy to configure, well-tested, and stable. The alternative ways to configure an RP are static RP, Auto-RP, and bootstrap router.

Sparse-Dense Mode for Auto-RP

A prerequisite of Auto-RP is that all interfaces must be configured in sparse-dense mode using the **ip pim sparse-dense-mode** interface configuration command. An interface configured in sparse-dense mode is treated in either sparse mode or dense mode of operation, depending on which mode the multicast group operates. If a multicast group has a known RP, the interface is treated in sparse mode. If a group has no known RP, by default the interface is treated in dense mode and data will be flooded over this interface. (You can prevent dense-mode fallback; see the module “Configuring Basic IP Multicast.”)

To successfully implement Auto-RP and prevent any groups other than 224.0.1.39 and 224.0.1.40 from operating in dense mode, we recommend configuring a “sink RP” (also known as “RP of last resort”). A sink RP is a statically configured RP that may or may not actually exist in the network. Configuring a sink RP does not interfere with Auto-RP operation because, by default, Auto-RP messages supersede static RP configurations. We recommend configuring a sink RP for all possible multicast groups in your network, because it is possible for an unknown or unexpected source to become active. If no RP is configured to limit source registration, the group may revert to dense mode operation and be flooded with data.

Bootstrap Router

Another RP selection model called bootstrap router (BSR) was introduced after Auto-RP in PIM-SM version 2. BSR performs similarly to Auto-RP in that it uses candidate routers for the RP function and for relaying the RP information for a group. RP information is distributed through BSR messages, which are carried within PIM messages. PIM messages are link-local multicast messages that travel from PIM router to PIM router. Because of this single hop method of disseminating RP information, TTL scoping cannot be used with BSR. A BSR performs similarly as an RP, except that it does not run the risk of reverting to dense mode operation, and it does not offer the ability to scope within a domain.

PIM Domain Border

As IP multicast becomes more widespread, the chance of one PIMv2 domain bordering another PIMv2 domain increases. Because two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. Allowing messages to leak across the domain borders could adversely affect the normal BSR election mechanism and elect a single BSR across all bordering domains and coningle candidate RP advertisements, resulting in the election of RPs in the wrong domain.

Multicast Forwarding

Forwarding of multicast traffic is accomplished by multicast-capable routers. These routers create distribution trees that control the path that IP multicast traffic takes through the network in order to deliver traffic to all receivers.

Multicast traffic flows from the source to the multicast group over a distribution tree that connects all of the sources to all of the receivers in the group. This tree may be shared by all sources (a shared tree) or a separate distribution tree can be built for each source (a source tree). The shared tree may be one-way or bidirectional.

Before describing the structure of source and shared trees, it is helpful to explain the notations that are used in multicast routing tables. These notations include the following:

- (S,G) = (unicast source for the multicast group G, multicast group G)
- (*,G) = (any source for the multicast group G, multicast group G)

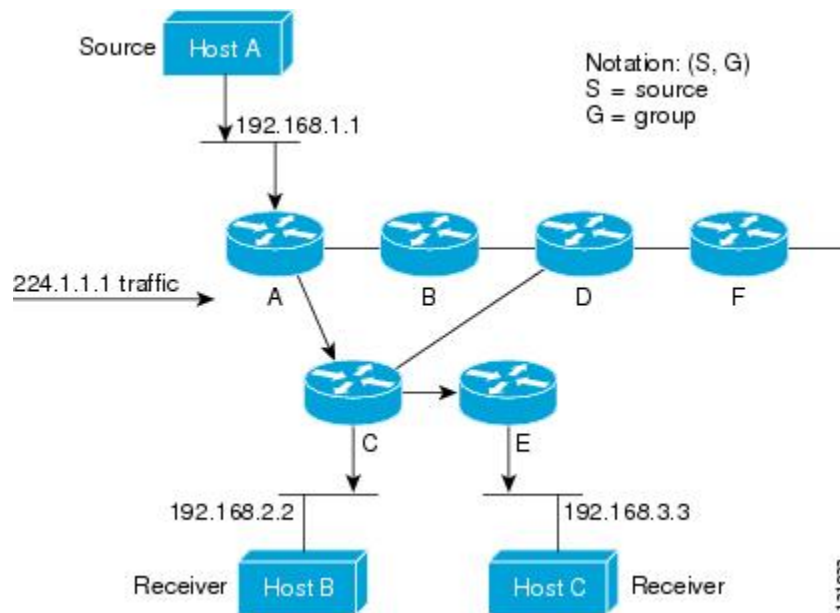
The notation of (S,G), pronounced “S comma G,” enumerates a shortest path tree where S is the IP address of the source and G is the multicast group address.

Shared trees are (*,G) and the source trees are (S,G) and always routed at the sources.

Multicast Distribution Source Tree

The simplest form of a multicast distribution tree is a source tree. A source tree has its root at the source host and has branches forming a spanning tree through the network to the receivers. Because this tree uses the shortest path through the network, it is also referred to as a shortest path tree (SPT).

The figure shows an example of an SPT for group 224.1.1.1 rooted at the source, Host A, and connecting two receivers, Hosts B and C.



Using standard notation, the SPT for the example shown in the figure would be (192.168.1.1, 224.1.1.1).

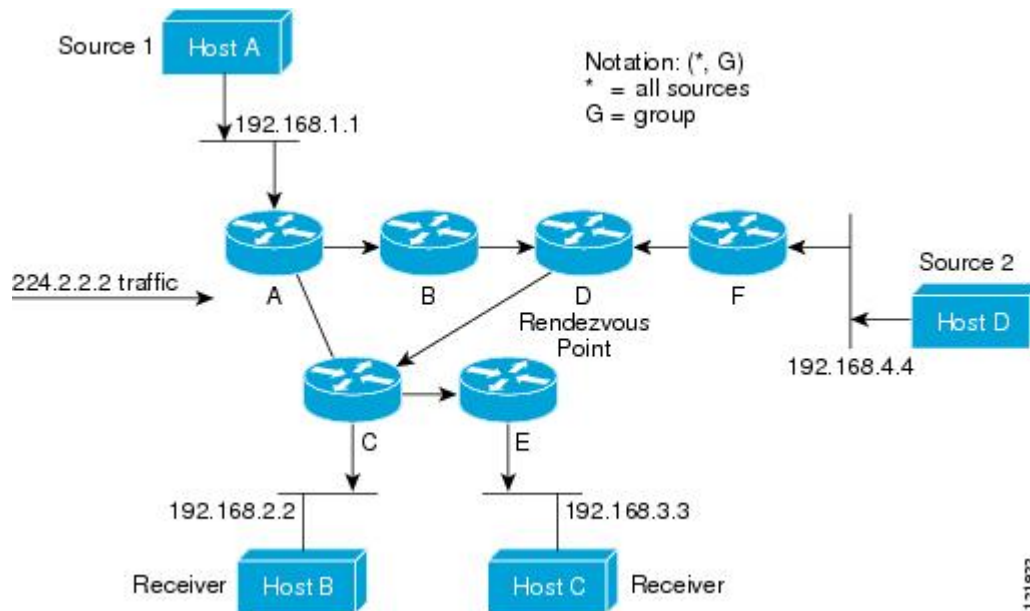
The (S,G) notation implies that a separate SPT exists for each individual source sending to each group--which is correct.

Multicast Distribution Shared Tree

Unlike source trees that have their root at the source, shared trees use a single common root placed at some chosen point in the network. This shared root is called a rendezvous point (RP).

The following figure shows a shared tree for the group 224.2.2.2 with the root located at Router D. This shared tree is unidirectional. Source traffic is sent towards the RP on a source tree. The traffic is then forwarded down the shared tree from the RP to reach all of the receivers (unless the receiver is located between the source and the RP, in which case it will be serviced directly).

Figure 7: Shared Tree



In this example, multicast traffic from the sources, Hosts A and D, travels to the root (Router D) and then down the shared tree to the two receivers, Hosts B and C. Because all sources in the multicast group use a common shared tree, a wildcard notation written as (*, G), pronounced “star comma G,” represents the tree. In this case, * means all sources, and G represents the multicast group. Therefore, the shared tree shown in the figure would be written as (*, 224.2.2.2).

Both source trees and shared trees are loop-free. Messages are replicated only where the tree branches. Members of multicast groups can join or leave at any time; therefore the distribution trees must be dynamically updated. When all the active receivers on a particular branch stop requesting the traffic for a particular multicast group, the routers prune that branch from the distribution tree and stop forwarding traffic down that branch. If one receiver on that branch becomes active and requests the multicast traffic, the router will dynamically modify the distribution tree and start forwarding traffic again.

Source Tree Advantage

Source trees have the advantage of creating the optimal path between the source and the receivers. This advantage guarantees the minimum amount of network latency for forwarding multicast traffic. However, this optimization comes at a cost. The routers must maintain path information for each source. In a network that has thousands of sources and thousands of groups, this overhead can quickly become a resource issue on the routers. Memory consumption from the size of the multicast routing table is a factor that network designers must take into consideration.

Shared Tree Advantage

Shared trees have the advantage of requiring the minimum amount of state in each router. This advantage lowers the overall memory requirements for a network that only allows shared trees. The disadvantage of shared trees is that under certain circumstances the paths between the source and receivers might not be the optimal paths, which might introduce some latency in packet delivery. For example, in the figure above the shortest path between Host A (source 1) and Host B (a receiver) would be Router A and Router C. Because we are using Router D as the root for a shared tree, the traffic must traverse Routers A, B, D and then C.

Network designers must carefully consider the placement of the rendezvous point (RP) when implementing a shared tree-only environment.

In unicast routing, traffic is routed through the network along a single path from the source to the destination host. A unicast router does not consider the source address; it considers only the destination address and how to forward the traffic toward that destination. The router scans through its routing table for the destination address and then forwards a single copy of the unicast packet out the correct interface in the direction of the destination.

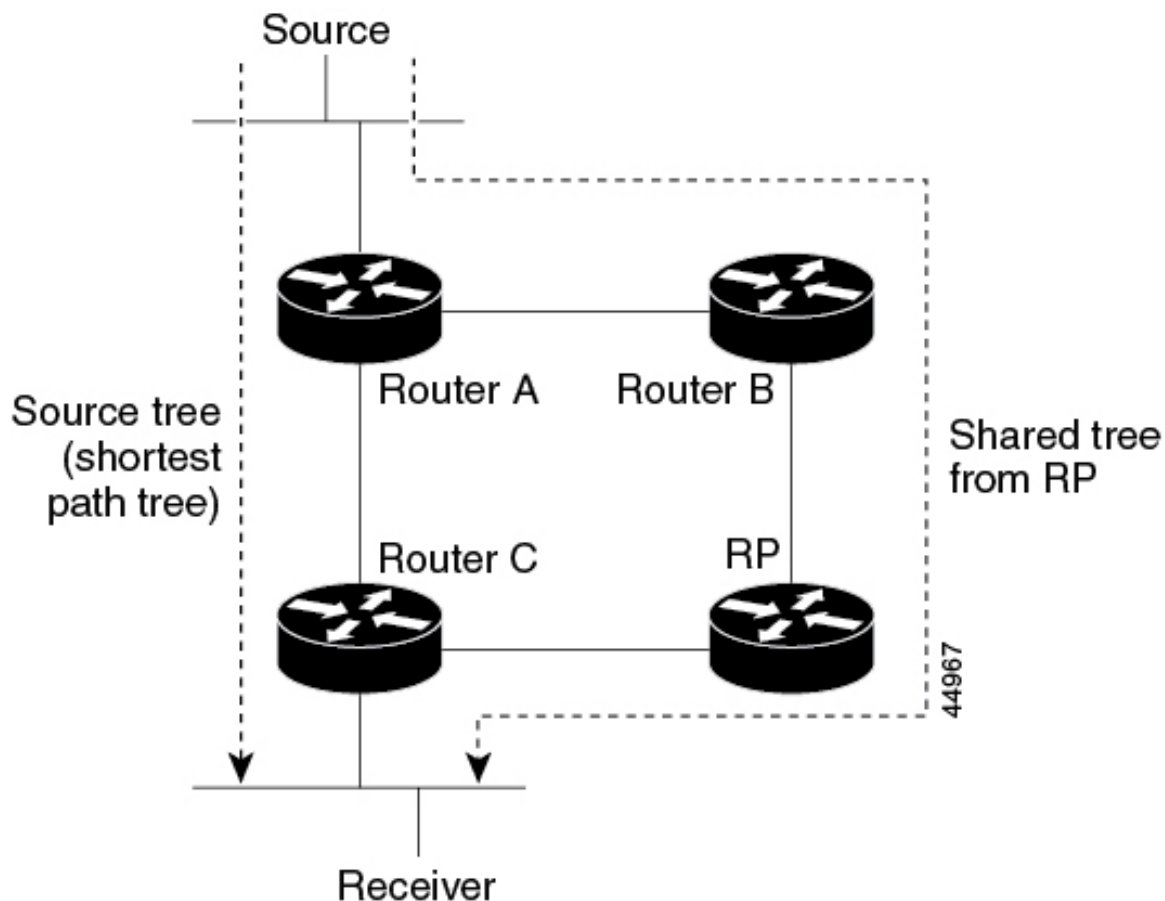
In multicast forwarding, the source is sending traffic to an arbitrary group of hosts that are represented by a multicast group address. The multicast router must determine which direction is the upstream direction (toward the source) and which one is the downstream direction (or directions) toward the receivers. If there are multiple downstream paths, the router replicates the packet and forwards it down the appropriate downstream paths (best unicast route metric)--which is not necessarily all paths. Forwarding multicast traffic away from the source, rather than to the receiver, is called Reverse Path Forwarding (RPF). RPF is described in the following section.

PIM Shared Tree and Source Tree

By default, members of a group receive data from senders to the group across a single data-distribution tree rooted at the RP.

Figure 8: Shared Tree and Source Tree (Shortest-Path Tree)

The following figure shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.



If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software devices to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP puts a link to Router C in its outgoing interface list.
3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.
4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a join message toward the source.
7. When Router C receives data on (S, G), it sends a prune message for the source up the shared tree.

8. The RP deletes the link to Router C from the outgoing interface of (S, G). The RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree. You can configure the PIM device to stay on the shared tree.

The change from shared to source tree happens when the first data packet arrives at the last-hop router. This change depends upon the threshold that is configured by using the **ip pim spt-threshold** global configuration command.

The shortest-path tree requires more memory than the shared tree but reduces delay. You may want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

Reverse Path Forwarding

In unicast routing, traffic is routed through the network along a single path from the source to the destination host. A unicast router does not consider the source address; it considers only the destination address and how to forward the traffic toward that destination. The router scans through its routing table for the destination network and then forwards a single copy of the unicast packet out the correct interface in the direction of the destination.

In multicast forwarding, the source is sending traffic to an arbitrary group of hosts that are represented by a multicast group address. The multicast router must determine which direction is the upstream direction (toward the source) and which one is the downstream direction (or directions) toward the receivers. If there are multiple downstream paths, the router replicates the packet and forwards it down the appropriate downstream paths (best unicast route metric)--which is not necessarily all paths. Forwarding multicast traffic away from the source, rather than to the receiver, is called Reverse Path Forwarding (RPF). RPF is an algorithm used for forwarding multicast datagrams.

Protocol Independent Multicast (PIM) uses the unicast routing information to create a distribution tree along the reverse path from the receivers towards the source. The multicast routers then forward packets along the distribution tree from the source to the receivers. RPF is a key concept in multicast forwarding. It enables routers to correctly forward multicast traffic down the distribution tree. RPF makes use of the existing unicast routing table to determine the upstream and downstream neighbors. A router will forward a multicast packet only if it is received on the upstream interface. This RPF check helps to guarantee that the distribution tree will be loop-free.

RPF Check

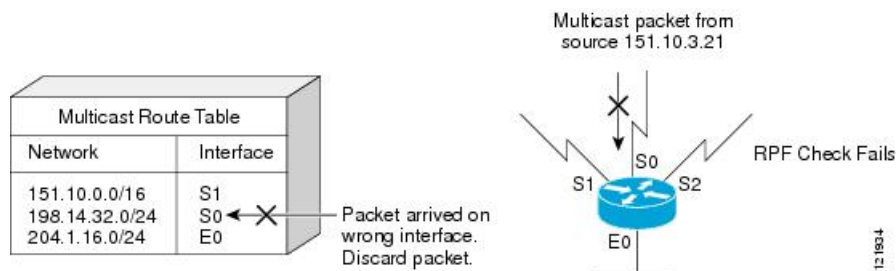
When a multicast packet arrives at a router, the router performs an RPF check on the packet. If the RPF check succeeds, the packet is forwarded. Otherwise, it is dropped.

For traffic flowing down a source tree, the RPF check procedure works as follows:

1. The router looks up the source address in the unicast routing table to determine if the packet has arrived on the interface that is on the reverse path back to the source.
2. If the packet has arrived on the interface leading back to the source, the RPF check succeeds and the packet is forwarded out the interfaces present in the outgoing interface list of a multicast routing table entry.
3. If the RPF check in Step 2 fails, the packet is dropped.

The figure shows an example of an unsuccessful RPF check.

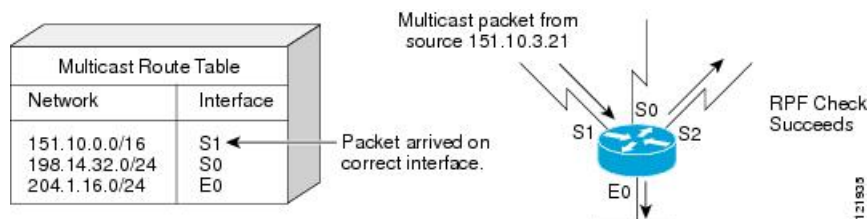
Figure 9: RPF Check Fails



As the figure illustrates, a multicast packet from source 151.10.3.21 is received on serial interface 0 (S0). A check of the unicast route table shows that S1 is the interface this router would use to forward unicast data to 151.10.3.21. Because the packet has arrived on interface S0, the packet is discarded.

The figure shows an example of a successful RPF check.

Figure 10: RPF Check Succeeds



In this example, the multicast packet has arrived on interface S1. The router refers to the unicast routing table and finds that S1 is the correct interface. The RPF check passes, and the packet is forwarded.

Default PIM Routing Configuration

This table displays the default PIM routing configuration for the device.

Table 27: Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM stub routing	None configured.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kb/s.
PIM router query message interval	30 seconds.

How to Configure PIM

Enabling PIM Stub Routing

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **ip pim passive**
5. **end**
6. **show ip pim interface**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies the interface on which you want to enable PIM stub routing, and enters interface configuration mode.
Step 4	ip pim passive Example: Switch(config-if)# ip pim passive	Configures the PIM stub feature on the interface.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show ip pim interface Example: Switch# show ip pim interface	(Optional) Displays the PIM stub that is enabled on each interface.
Step 7	show running-config Example: Switch# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Configuring a Rendezvous Point

You must have a rendezvous point (RP), if the interface is in sparse-dense mode and if you want to handle the group as a sparse group. You can use these methods:

- By manually assigning an RP to multicast groups.
- As a standalone, Cisco-proprietary protocol separate from PIMv1, which includes:
 - Setting up Auto-RP in a new internetwork
 - Adding Auto-RP to an existing sparse-mode cloud
 - Preventing join messages to false RPs
 - Filtering incoming RP announcement messages
- By using a standards track protocol in the Internet Engineering Task Force (IETF), which includes configuring PIMv2 BSR .



Note You can use Auto-RP, BSR, or a combination of both, depending on the PIM version that you are running and the types of routers in your network. For information about working with different PIM versions in your network, see the PIMv1 and PIMv2 Interoperability section.

Manually Assigning an RP to Multicast Groups

If the rendezvous point (RP) for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you need not perform this task for that RP.

Senders of multicast traffic announce their existence through register messages received from the source first-hop router (designated router) and forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages.



Note RPs are not members of the multicast group; they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer device responds to the group as dense and uses the dense-mode PIM techniques.

This procedure is optional.

SUMMARY STEPS

1. `enable`

2. **configure terminal**
3. **ip pim rp-address** *ip-address* [*access-list-number*] [**override**]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip pim rp-address <i>ip-address</i> [<i>access-list-number</i>] [override] Example: Switch(config)# ip pim rp-address 10.1.1.1 20 override	Configures the address of a PIM RP. By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer devices (including the RP). Note If there is no RP configured for a group, the device treats the group as dense, using the dense-mode PIM techniques. A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access list conditions specify for which groups the device is an RP. <ul style="list-style-type: none"> • For <i>ip-address</i>, enter the unicast address of the RP in dotted-decimal notation. • (Optional) For <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • (Optional) The override keyword indicates that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails.
Step 4	access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>] Example:	Creates a standard access list, repeating the command as many times as necessary.

	Command or Action	Purpose
	<pre>Switch(config)# access-list 25 permit 10.5.0.1 255.224.0.0</pre>	<ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 2. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the multicast group address for which the RP should be used. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting Up Auto-RP in a New Internetwork

If you are setting up Auto-RP in a new internetwork, you do not need a default RP because you configure all the interfaces for sparse-dense mode.



Note Omit Step 3 in the following procedure, if you want to configure a PIM router as the RP for the local group.

SUMMARY STEPS

1. enable

2. **show running-config**
3. **configure terminal**
4. **ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*
5. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
6. **ip pim send-rp-discovery** *scope* *ttl*
7. **end**
8. **show running-config**
9. **show ip pim rp mapping**
10. **show ip pim rp**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show running-config Example: <pre>Switch# show running-config</pre>	Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the ip pim rp-address global configuration command. <p>Note This step is not required for sparse-dense-mode environments.</p> <p>The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.</p>
Step 3	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 4	ip pim send-rp-announce <i>interface-id</i> scope <i>ttl</i> group-list <i>access-list-number</i> interval <i>seconds</i> Example: <pre>Switch(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120</pre>	Configures another PIM device to be the candidate RP for local groups. <ul style="list-style-type: none"> • For <i>interface-id</i>, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. • For scope <i>ttl</i>, specify the time-to-live value in hops. Enter a hop count that is high enough so that the

	Command or Action	Purpose
		<p>RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255.</p> <ul style="list-style-type: none"> For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. For interval <i>seconds</i>, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.
Step 5	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 10 permit 10.10.0.0</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the multicast group address range for which the RP should be used. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Note Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 6	<p>ip pim send-rp-discovery scope <i>ttl</i></p> <p>Example:</p> <pre>Switch(config)# ip pim send-rp-discovery scope 50</pre>	<p>Finds a device whose connectivity is not likely to be interrupted, and assign it the role of RP-mapping agent.</p> <p>For scope <i>ttl</i>, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
Step 8	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 9	show ip pim rp mapping Example: Switch# <code>show ip pim rp mapping</code>	Displays active RPs that are cached with associated multicast routing entries.
Step 10	show ip pim rp Example: Switch# <code>show ip pim rp</code>	Displays the information cached in the routing table.
Step 11	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Adding Auto-RP to an Existing Sparse-Mode Cloud

This section contains suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `show running-config`
3. `configure terminal`
4. `ip pim send-rp-announce interface-id scope ttl group-list access-list-number interval seconds`
5. `access-list access-list-number {deny | permit} source [source-wildcard]`
6. `ip pim send-rp-discovery scope ttl`
7. `end`
8. `show running-config`
9. `show ip pim rp mapping`
10. `show ip pim rp`
11. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the ip pim rp-address global configuration command.</p> <p>Note This step is not required for sparse-dense-mode environments.</p> <p>The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.</p>
Step 3	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 4	<p>ip pim send-rp-announce <i>interface-id</i> scope <i>ttl</i> group-list <i>access-list-number</i> interval <i>seconds</i></p> <p>Example:</p> <pre>Switch(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120</pre>	<p>Configures another PIM device to be the candidate RP for local groups.</p> <ul style="list-style-type: none"> • For <i>interface-id</i>, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. • For scope <i>ttl</i>, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. • For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • For interval <i>seconds</i>, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.

	Command or Action	Purpose
Step 5	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 10 permit 224.0.0.0 15.255.255.255</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the multicast group address range for which the RP should be used. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 6	<p>ip pim send-rp-discovery scope <i>ttl</i></p> <p>Example:</p> <pre>Switch(config)# ip pim send-rp-discovery scope 50</pre>	<p>Finds a device whose connectivity is not likely to be interrupted, and assigns it the role of RP-mapping agent.</p> <p>For scope <i>ttl</i>, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.</p> <p>Note To remove the device as the RP-mapping agent, use the no ip pim send-rp-discovery global configuration command.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>
Step 9	<p>show ip pim rp mapping</p> <p>Example:</p>	<p>Displays active RPs that are cached with associated multicast routing entries.</p>

	Command or Action	Purpose
	Switch# show ip pim rp mapping	
Step 10	show ip pim rp Example: Switch# show ip pim rp	Displays the information cached in the routing table.
Step 11	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Sparse Mode with a Single Static RP(CLI)

A rendezvous point (RP) is required in networks running Protocol Independent Multicast sparse mode (PIM-SM). In PIM-SM, traffic will be forwarded only to network segments with active receivers that have explicitly requested multicast data.

This section describes how to configure sparse mode with a single static RP.

Before you begin

All access lists that are needed when sparse mode is configured with a single static RP should be configured prior to beginning the configuration task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing [distributed]**
4. **interface** *type number*
5. **ip pim sparse-mode**
6. Repeat Steps 1 through 5 on every interface that uses IP multicast.
7. **exit**
8. **ip pim rp-address** *rp-address* [*access-list*] [**override**]
9. **end**
10. **show ip pim rp [mapping] [rp-address]**
11. **show ip igmp groups** [*group-name* | *group-address*] [*interface-type interface-number*] [**detail**]
12. **show ip mroute**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: <pre>device> enable</pre>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: <pre>device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip multicast-routing [distributed] Example: <pre>device(config)# ip multicast-routing</pre>	Enables IP multicast routing. <ul style="list-style-type: none"> Use the distributed keyword to enable Multicast Distributed Switching.
Step 4	interface type number Example: <pre>device(config)# interface gigabitethernet 1/0/0</pre>	Selects an interface that is connected to hosts on which PIM can be enabled.
Step 5	ip pim sparse-mode Example: <pre>device(config-if)# ip pim sparse-mode</pre>	Enables PIM on an interface. You must use sparse mode.
Step 6	Repeat Steps 1 through 5 on every interface that uses IP multicast.	--
Step 7	exit Example: <pre>device(config-if)# exit</pre>	Returns to global configuration mode.
Step 8	ip pim rp-address rp-address [access-list] [override] Example: <pre>device(config)# ip pim rp-address 192.168.0.0</pre>	Configures the address of a PIM RP for a particular group. <ul style="list-style-type: none"> The optional <i>access-list</i> argument is used to specify the number or name a standard access list that defines the multicast groups to be statically mapped to the RP. <p>Note If no access list is defined, the RP will map to all multicast groups, 224/4.</p> <ul style="list-style-type: none"> The optional override keyword is used to specify that if dynamic and static group-to-RP mappings are used together and there is an RP address conflict, the RP address configured for a static group-to-RP mapping will take precedence.

	Command or Action	Purpose
		Note If the override keyword is not specified and there is RP address conflict, dynamic group-to-RP mappings will take precedence over static group-to-RP mappings.
Step 9	end Example: device(config)# end	Ends the current configuration session and returns to EXEC mode.
Step 10	show ip pim rp [mapping] [rp-address] Example: device# show ip pim rp mapping	(Optional) Displays RPs known in the network and shows how the router learned about each RP.
Step 11	show ip igmp groups [group-name group-address interface-type interface-number] [detail] Example: device# show ip igmp groups	(Optional) Displays the multicast groups having receivers that are directly connected to the router and that were learned through IGMP. <ul style="list-style-type: none"> • A receiver must be active on the network at the time that this command is issued in order for receiver information to be present on the resulting display.
Step 12	show ip mroute Example: device# show ip mroute	(Optional) Displays the contents of the IP mroute table.

Preventing Join Messages to False RPs

Determine whether the **ip pim accept-rp** command was previously configured throughout the network by using the **show running-config** privileged EXEC command. If the **ip pim accept-rp** command is not configured on any device, this problem can be addressed later. In those routers or multilayer devices already configured with the **ip pim accept-rp** command, you must enter the command again to accept the newly advertised RP.

To accept all RPs advertised with Auto-RP and reject all other RPs by default, use the **ip pim accept-rp auto-rp** global configuration command.

This procedure is optional.

Filtering Incoming RP Announcement Messages

You can add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number*
4. **access-list** *access-list-number* {deny | permit} *source* [*source-wildcard*]
5. end
6. show running-config
7. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip pim rp-announce-filter rp-list <i>access-list-number</i> group-list <i>access-list-number</i></p> <p>Example:</p> <pre>Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 14</pre>	<p>Filters incoming RP announcement messages.</p> <p>Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default.</p> <p>For rp-list <i>access-list-number</i>, configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the group-list <i>access-list-number</i> variable. If this variable is omitted, the filter applies to all multicast groups.</p> <p>If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the group-to-RP mapping information.</p>
Step 4	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 10 permit 10.8.1.0 255.255.224.0</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 2. • The deny keyword denies access if the conditions are matched. • The permit keyword permits access if the conditions are matched. • Create an access list that specifies from which routers and multilayer devices the mapping agent accepts candidate RP announcements (rp-list ACL).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL). • For <i>source</i>, enter the multicast group address range for which the RP should be used. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring PIMv2 BSR

The process for configuring PIMv2 BSR may involve the following optional tasks:

- Defining the PIM domain border
- Defining the IP multicast boundary
- Configuring candidate BSRs
- Configuring candidate RPs

Defining the PIM Domain Border

Perform the following steps to configure the PIM domain border. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim bsr-border**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip pim bsr-border Example: <pre>Switch(config-if)# ip pim bsr-border</pre>	Defines a PIM bootstrap message boundary for the PIM domain. Enter this command on each interface that connects to other bordering PIM domains. This command instructs the device to neither send nor receive PIMv2 BSR messages on this interface. Note To remove the PIM border, use the no ip pim bsr-border interface configuration command.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show running-config</code>	
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `access-list access-list-number deny source [source-wildcard]`
4. `interface interface-id`
5. `ip multicast boundary access-list-number`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	access-list access-list-number deny source [source-wildcard] Example: Switch(config)# <code>access-list 12 deny 224.0.1.39</code>	Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched.

	Command or Action	Purpose
	<code>access-list 12 deny 224.0.1.40</code>	<ul style="list-style-type: none"> For <i>source</i>, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 4	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface to be configured, and enters interface configuration mode.
Step 5	ip multicast boundary <i>access-list-number</i> Example: <pre>Switch(config-if)# ip multicast boundary 12</pre>	Configures the boundary, specifying the access list you created in Step 2.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Candidate BSRs

You can configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip pim bsr-candidate <i>interface-id hash-mask-length</i> [<i>priority</i>] Example: <pre>Switch(config)# ip pim bsr-candidate gigabitethernet 1/0/3 28 100</pre>	Configures your device to be a candidate BSR. <ul style="list-style-type: none"> • For <i>interface-id</i>, enter the interface on this device from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs. • For <i>hash-mask-length</i>, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the hash function is called. All groups with the same seed hash correspond to the same RP. For example, if this value is 24, only the first 24 bits of the group addresses matter. • (Optional) For <i>priority</i>, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the Candidate RPs

You can configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP can serve the entire IP multicast address space or a portion of it. Candidate RPs send candidate RP advertisements to the BSR.

This procedure is optional.

Before you begin

When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer devices where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer devices and with routers from other vendors, any device can be used as an RP.
- In a network of Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure only Cisco PIMv2 routers and multilayer devices as RPs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> <code>enable</code>	
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip pim rp-candidate <i>interface-id</i> [group-list <i>access-list-number</i>] Example: Switch(config)# <code>ip pim rp-candidate gigabitethernet 1/0/5 group-list 10</code>	Configures your device to be a candidate RP. <ul style="list-style-type: none"> For <i>interface-id</i>, specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs. (Optional) For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no group-list is specified, the device is a candidate RP for all groups.
Step 4	access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>] Example: Switch(config)# <code>access-list 10 permit 239.0.0.0 0.255.255.255</code>	Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 2. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the number of the network or host from which the packet is being sent. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show running-config</code>	
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Delaying the Use of PIM Shortest-Path Tree

Perform these steps to configure a traffic rate threshold that must be reached before multicast routing is switched from the source tree to the shortest-path tree.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `access-list access-list-number {deny | permit} source [source-wildcard]`
4. `ip pim spt-threshold {kpbs | infinity} [group-list access-list-number]`
5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	access-list access-list-number {deny permit} source [source-wildcard] Example: Switch(config)# <code>access-list 16 permit 225.0.0.0</code>	Creates a standard access list. <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched.

	Command or Action	Purpose
	0.255.255.255	<ul style="list-style-type: none"> The permit keyword permits access if the conditions are matched. For <i>source</i>, specify the multicast group to which the threshold will apply. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 4	<p>ip pim spt-threshold {<i>kbps</i> infinity} [group-list <i>access-list-number</i>]</p> <p>Example:</p> <pre>Switch(config)# ip pim spt-threshold infinity group-list 16</pre>	<p>Specifies the threshold that must be reached before moving to shortest-path tree (spt).</p> <ul style="list-style-type: none"> For <i>kbps</i>, specify the traffic rate in kilobits per second. The default is 0 kbps. <p>Note Because of device hardware limitations, 0 kbps is the only valid entry even though the range is 0 to 4294967.</p> <ul style="list-style-type: none"> Specify infinity if you want all sources for the specified group to use the shared tree, never switching to the source tree. (Optional) For group-list <i>access-list-number</i>, specify the access list created in Step 2. If the value is 0 or if the group list is not used, the threshold applies to all groups.
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Modifying the PIM Router-Query Message Interval

PIM routers and multilayer devices send PIM router-query messages to find which device will be the designated router (DR) for each LAN segment (subnet). The DR is responsible for sending IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM-SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim query-interval** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip pim query-interval <i>seconds</i> Example: <pre>Switch(config-if)# ip pim</pre>	Configures the frequency at which the device sends PIM router-query messages. The default is 30 seconds. The range is 1 to 65535.

	Command or Action	Purpose
	<code>query-interval 45</code>	
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: <pre>Switch# show ip igmp interface</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Verifying PIM Operations

Verifying IP Multicast Operation in a PIM-SM or a PIM-SSM Network

Perform the following optional tasks to verify IP multicast operation in a PIM-SM or a PIM-SSM network. The steps in these tasks help to locate a faulty hop when sources and receivers are not operating as expected.



Note If packets are not reaching their expected destinations, you might want consider disabling IP multicast fast switching, which would place the router in process switching mode. If packets begin reaching their proper destinations after IP multicast fast switching has been disabled, then the issue most likely was related to IP multicast fast switching.

Verifying IP Multicast on the First Hop Router

Enter these commands on the first hop router to verify IP multicast operations on the first hop router:

SUMMARY STEPS

1. `enable`
2. `show ip mroute` [*group-address*]
3. `show ip mroute active` [*kb/s*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip mroute [<i>group-address</i>] Example: Switch# show ip mroute 239.1.2.3 (*, 239.1.2.3), 00:18:10/stopped, RP 172.16.0.1, flags: SPF Incoming interface: Serial1/0, RPF nbr 172.31.200.2 Outgoing interface list: Null (10.0.0.1, 239.1.2.3), 00:18:10/00:03:22, flags: FT Incoming interface: GigabitEthernet0/0/0, RPF nbr 0.0.0.0 Outgoing interface list: Serial1/0, Forward/Sparse-Dense, 00:18:10/00:03:19	Confirms that the F flag has been set for mroutes on the first hop router.
Step 3	show ip mroute active [<i>kb/s</i>] Example: Switch# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)	Displays information about active multicast sources sending to groups. The output of this command provides information about the multicast packet rate for active sources. Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.

Verifying IP Multicast on Routers Along the SPT

Enter these commands on routers along the SPT to verify IP multicast operations on routers along the SPT in a PIM-SM or PIM-SSM network:

SUMMARY STEPS

1. **enable**

2. `show ip mroute [group-address]`
3. `show ip mroute active`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>show ip mroute [group-address]</p> <p>Example:</p> <pre>Switch# show ip mroute 239.1.2.3 (*, 239.1.2.3), 00:17:56/00:03:02, RP 172.16.0.1, flags: S Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse-Dense, 00:17:56/00:03:02 (10.0.0.1, 239.1.2.3), 00:15:34/00:03:28, flags: T Incoming interface: Serial1/0, RPF nbr 172.31.200.1 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse-Dense, 00:15:34/00:03:02</pre>	<p>Confirms the RPF neighbor towards the source for a particular group or groups.</p>
Step 3	<p>show ip mroute active</p> <p>Example:</p> <pre>Switch# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)</pre>	<p>Displays information about active multicast sources sending to groups. The output of this command provides information about the multicast packet rate for active sources.</p> <p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p>

Verifying IP Multicast Operation on the Last Hop Router

Enter these commands on the last hop router to verify IP multicast operations on the last hop router:

SUMMARY STEPS

1. enable
2. show ip igmp groups
3. show ip pim rp mapping
4. show ip mroute
5. show ip interface [type number]
6. show ip pim interface count
7. show ip mroute count
8. show ip mroute active [kb/s]

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>show ip igmp groups</p> <p>Example:</p> <pre>Switch# show ip igmp groups IGMP Connected Group Membership Group Address Interface Uptime Expires Last Reporter 239.1.2.3 GigabitEthernet1/0/0 00:05:14 00:02:14 10.1.0.6 224.0.1.39 GigabitEthernet0/0/0 00:09:11 00:02:08 172.31.100.1</pre>	<p>Verifies IGMP memberships on the last hop router. This information will confirm the multicast groups with receivers that are directly connected to the last hop router and that are learned through IGMP.</p>
Step 3	<p>show ip pim rp mapping</p> <p>Example:</p> <pre>Switch# show ip pim rp mapping PIM Group-to-RP Mappings Group(s) 224.0.0.0/4 RP 172.16.0.1 (?), v2v1 Info source: 172.16.0.1 (?), elected via Auto-RP Uptime: 00:09:11, expires: 00:02:47</pre>	<p>Confirms that the group-to-RP mappings are being populated correctly on the last hop router.</p> <p>Note Ignore this step if you are verifying a last hop router in a PIM-SSM network. The show ip pim rp mapping command does not work with routers in a PIM-SSM network because PIM-SSM does not use RPs. In addition, if configured correctly, PIM-SSM groups do not appear in the output of the show ip pim rp mapping command.</p>
Step 4	<p>show ip mroute</p> <p>Example:</p> <pre>Switch# show ip mroute (*, 239.1.2.3), 00:05:14/00:03:04, RP 172.16.0.1, flags: SJC Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense,</pre>	<p>Verifies that the mroute table is being populated properly on the last hop router.</p>

	Command or Action	Purpose
	<pre>00:05:10/00:03:04 (10.0.0.1, 239.1.2.3), 00:02:49/00:03:29, flags: T Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense, 00:02:49/00:03:04 (*, 224.0.1.39), 00:10:05/stopped, RP 0.0.0.0, flags: DC Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense, 00:05:15/00:00:00 GigabitEthernet0/0, Forward/Sparse-Dense, 00:10:05/00:00:00 (172.16.0.1, 224.0.1.39), 00:02:00/00:01:33, flags: PTX Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1</pre>	
Step 5	<p>show ip interface [<i>type number</i>]</p> <p>Example:</p> <pre>Switch# show ip interface GigabitEthernet 0/0/0 GigabitEthernet0/0 is up, line protocol is up Internet address is 172.31.100.2/24 Broadcast address is 255.255.255.255 Address determined by setup command MTU is 1500 bytes Helper address is not set Directed broadcast forwarding is disabled Multicast reserved groups joined: 224.0.0.1 224.0.0.22 224.0.0.13 224.0.0.5 224.0.0.6 Outgoing access list is not set Inbound access list is not set Proxy ARP is enabled Local Proxy ARP is disabled Security level is default Split horizon is enabled ICMP redirects are always sent ICMP unreachable are always sent ICMP mask replies are never sent IP fast switching is enabled IP fast switching on the same interface is disabled IP Flow switching is disabled IP CEF switching is disabled IP Fast switching turbo vector IP multicast fast switching is enabled IP multicast distributed fast switching is disabled IP route-cache flags are Fast Router Discovery is disabled IP output packet accounting is disabled IP access violation accounting is disabled TCP/IP header compression is disabled RTP/IP header compression is disabled</pre>	<p>Verifies that multicast fast switching is enabled for optimal performance on the outgoing interface on the last hop router.</p> <p>Note Using the no ip mroute-cache interface command disables IP multicast fast-switching. When IP multicast fast switching is disabled, packets are forwarded through the process-switched path.</p>

	Command or Action	Purpose
	<pre> Policy routing is disabled Network address translation is disabled WCCP Redirect outbound is disabled WCCP Redirect inbound is disabled WCCP Redirect exclude is disabled BGP Policy Mapping is disabled </pre>	
Step 6	<p>show ip pim interface count</p> <p>Example:</p> <pre> Switch# show ip pim interface count State: * - Fast Switched, D - Distributed Fast Switched H - Hardware Switching Enabled Address Interface FS Mpackets In/Out 172.31.100.2 GigabitEthernet0/0/0 * 4122/0 10.1.0.1 GigabitEthernet1/0/0 * 0/3193 </pre>	Confirms that multicast traffic is being forwarded on the last hop router.
Step 7	<p>show ip mroute count</p> <p>Example:</p> <pre> Switch# show ip mroute count IP Multicast Statistics 6 routes using 4008 bytes of memory 3 groups, 1.00 average sources per group Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc) Group: 239.1.2.3, Source count: 1, Packets forwarded: 3165, Packets received: 3165 RP-tree: Forwarding: 0/0/0/0, Other: 0/0/0 Source: 10.0.0.1/32, Forwarding: 3165/20/28/4, Other: 0/0/0 Group: 224.0.1.39, Source count: 1, Packets forwarded: 21, Packets received: 120 Source: 172.16.0.1/32, Forwarding: 21/1/48/0, Other: 120/0/99 Group: 224.0.1.40, Source count: 1, Packets forwarded: 10, Packets received: 10 Source: 172.16.0.1/32, Forwarding: 10/1/48/0, Other: 10/0/0 </pre>	Confirms that multicast traffic is being forwarded on the last hop router.
Step 8	<p>show ip mroute active [kb/s]</p> <p>Example:</p> <pre> Switch# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps </pre>	Displays information about active multicast sources sending traffic to groups on the last hop router. The output of this command provides information about the multicast packet rate for active sources.

	Command or Action	Purpose
	<pre>Group: 239.1.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 50 secs), 4 kbps(life avg)</pre>	<p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p>

Using PIM-Enabled Routers to Test IP Multicast Reachability

If all the PIM-enabled routers and access servers that you administer are members of a multicast group, pinging that group causes all routers to respond, which can be a useful administrative and debugging tool.

To use PIM-enabled routers to test IP multicast reachability, perform the following tasks:

Configuring Routers to Respond to Multicast Pings

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip igmp join-group** *group-address*
5. Repeat Step 3 and Step 4 for each interface on the router participating in the multicast network.
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example:</p> <pre>Switch> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	<pre>configure terminal</pre> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<pre>interface type number</pre> <p>Example:</p>	Enters interface configuration mode.

Pinging Routers Configured to Respond to Multicast Pings

	Command or Action	Purpose
	Switch(config)# interface gigabitethernet 1/0/0	For the <i>type</i> and <i>number</i> arguments, specify an interface that is directly connected to hosts or is facing hosts.
Step 4	ip igmp join-group <i>group-address</i> Example: Switch(config-if)# ip igmp join-group 225.2.2.2	(Optional) Configures an interface on the router to join the specified group. For the purpose of this task, configure the same group address for the <i>group-address</i> argument on all interfaces on the router participating in the multicast network. Note With this method, the router accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the router from fast switching.
Step 5	Repeat Step 3 and Step 4 for each interface on the router participating in the multicast network.	--
Step 6	end Example: Switch(config-if)# end	Ends the current configuration session and returns to privileged EXEC mode.

Pinging Routers Configured to Respond to Multicast Pings

on a router to initiate a ping test to the routers configured to respond to multicast pings. This task is used to test IP multicast reachability in a network.

SUMMARY STEPS

1. **enable**
2. **ping** *group-address*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	ping <i>group-address</i> Example: Switch# ping 225.2.2.2	Pings an IP multicast group address. A successful response indicates that the group address is functioning.

Monitoring and Troubleshooting PIM

Monitoring PIM Information

Use the privileged EXEC commands in the following table to monitor your PIM configurations.

Table 28: PIM Monitoring Commands

Command	Purpose
<code>show ip pim interface</code>	Displays information about interfaces configured for Protocol Independent Multicast (PIM).
<code>show ip pim neighbor</code>	Displays the PIM neighbor information.
<code>show ip pim rp[group-name group-address]</code>	Displays RP routers associated with a sparse-mode multicast group. This command is available in all software images.

Monitoring the RP Mapping and BSR Information

Use the privileged EXEC mode in the following table to verify the consistency of group-to-RP mappings:

Table 29: RP Mapping Monitoring Commands

Command	Purpose
<code>show ip pim rp-hash group</code>	Displays the RP that was selected for the specified group. That is, on a PIMv2 router or multilayer device, confirms that the same RP is the one that a PIMv1 system chooses. For <i>group</i> , enter the group address for which to display RP information.

Use the privileged EXEC commands in the following table to monitor BSR information:

Table 30: BSR Monitoring Commands

Command	Purpose
<code>show ip pim bsr</code>	Displays information about the elected BSR.

Troubleshooting PIMv1 and PIMv2 Interoperability Problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the `show ip pim rp-hash` privileged EXEC command, making sure that all systems agree on the same RP for the same group.

2. Verify interoperability between different versions of DRs and RPs. Make sure that the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

Configuration Examples for PIM

Example: Enabling PIM Stub Routing

In this example, IP multicast routing is enabled, Switch A PIM uplink port 25 is configured as a routed uplink port with **spare-dense-mode** enabled. PIM stub routing is enabled on the VLAN 100 interfaces and on Gigabit Ethernet port 20.

```
Switch(config)# ip multicast-routing distributed
Switch(config)# interface GigabitEthernet3/0/25
Switch(config-if)# no switchport
Switch(config-if)# ip address 3.1.1.2 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet3/0/20
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip address 100.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet3/0/20
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# end
```

Example: Verifying PIM Stub Routing

To verify that PIM stub is enabled for each interface, use the **show ip pim interface** privileged EXEC command:

```
Switch# show ip pim interface
Address Interface Ver/ Nbr Query DR DR
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet3/0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet3/0/20 v2/P 0 30 1 10.1.1.1
```

Example: Manually Assigning an RP to Multicast Groups

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Switch(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Switch(config)# ip pim rp-address 147.106.6.22 1
```

Example: Configuring Auto-RP

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this device serves as RP:

```
Switch(config)# ip pim send-rp-announce gigabitethernet1/0/1 scope 31 group-list 5
Switch(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

Example: Defining the IP Multicast Boundary to Deny Auto-RP Information

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Switch(config)# access-list 1 deny 224.0.1.39
Switch(config)# access-list 1 deny 224.0.1.40
Switch(config)# access-list 1 permit all
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip multicast boundary 1
```

Example: Filtering Incoming RP Announcement Messages

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Switch(config)# access-list 10 permit host 172.16.5.1
Switch(config)# access-list 10 permit host 172.16.2.1
Switch(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Switch(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

The mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

Example: Preventing Join Messages to False RPs

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Switch(config)# ip pim accept-rp 172.10.20.1 1
```

Example: Configuring Candidate BSRs

```
Switch(config)# access-list 1 permit 224.0.1.39
Switch(config)# access-list 1 permit 224.0.1.40
```

Example: Configuring Candidate BSRs

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10.

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip address 172.21.24.18 255.255.255.0
Switch(config-if)# ip pim sparse-mode
Switch(config-if)# ip pim bsr-candidate gigabitethernet1/0/2 30 10
```

Example: Configuring Candidate RPs

This example shows how to configure the device to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Switch(config)# ip pim rp-candidate gigabitethernet1/0/2 group-list 4
Switch(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```



CHAPTER 16

Configuring HSRP Aware PIM

- [HSRP Aware PIM, on page 221](#)

HSRP Aware PIM

This module describes how to configure the HSRP Aware PIM feature for enabling multicast traffic to be forwarded through the Hot Standby Router Protocol (HSRP) active router (AR), allowing Protocol Independent Multicast (PIM) to leverage HSRP redundancy, avoid potential duplicate traffic, and enable failover.

Restrictions for HSRP Aware PIM

- HSRP IPv6 is not supported.
- Stateful failover is not supported. During PIM stateless failover, the HSRP group's virtual IP address transfers to the standby router but no mrouting state information is transferred. PIM listens and responds to state change events and creates mroute states upon failover.
- The maximum number of HSRP groups that can be tracked by PIM on each interface is 16.
- The redundancy priority for a PIM DR must be greater than the configured or default value (1) of the PIM DR priority on any device for which the same HSRP group is enabled or the HSRP Active will fail to win the DR election.
- Dense mode is not supported.
- HSRP address as PIM RP is not supported. HSRP aware PIM is for coordinating PIM DR election and HSRP primary election.

Information About HSRP Aware PIM

HSRP

Hot Standby Router Protocol (HSRP) is a Cisco proprietary redundancy protocol for establishing a fault-tolerant default gateway.

The protocol establishes a framework between network devices in order to achieve default gateway failover if the primary gateway becomes inaccessible. By sharing an IP address and a MAC (Layer 2) address, two or more devices can act as a single virtual router. The members of a virtual router group continually exchange

status messages and one device can assume the routing responsibility of another, should it go out of commission for either planned or unplanned reasons. Hosts continue to forward IP packets to a consistent IP and MAC address, and the changeover of devices doing the routing is transparent.

HSRP is useful for hosts that do not support a router discovery protocol and cannot switch to a new device when their selected device reloads or loses power. Because existing TCP sessions can survive the failover, this protocol also provides a more transparent recovery for hosts that dynamically choose a next hop for routing IP traffic.

When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among a group of devices running HSRP. The address of this HSRP group is referred to as the virtual IP address. One of these devices is selected by the protocol to be the active router (AR). The AR receives and routes packets destined for the MAC address of the group.

HSRP uses a priority mechanism to determine which HSRP configured device is to be the default AR. To configure a device as the AR, you assign it a priority that is higher than the priority of all the other HSRP-configured devices. The default priority is 100, so if you configure just one device to have a higher priority, that device will be the default AR.

Devices that are running HSRP send and receive multicast User Datagram Protocol (UDP)-based hello messages to detect device failure and to designate active and standby devices. When the AR fails to send a hello message within a configurable period of time, the standby device with the highest priority becomes the AR. The transition of packet forwarding functions between devices is completely transparent to all hosts on the network.

You can configure multiple Hot Standby groups on an interface, thereby making fuller use of redundant devices and load sharing.

HSRP is not a routing protocol as it does not advertise IP routes or affect the routing table in any way.

HSRP has the ability to trigger a failover if one or more interfaces on the device fail. This can be useful for dual branch devices each with a single serial link back to the head end. If the serial link of the primary device goes down, the backup device takes over the primary functionality and thus retains connectivity to the head end.

HSRP Aware PIM

Protocol Independent Multicast (PIM) has no inherent redundancy capabilities and its operation is completely independent of Hot Standby Router Protocol (HSRP) group states. As a result, IP multicast traffic is forwarded not necessarily by the same device as is elected by HSRP. The HSRP Aware PIM feature provides consistent IP multicast forwarding in a redundant network with virtual routing groups enabled.

HSRP Aware PIM enables multicast traffic to be forwarded through the HSRP active router (AR), allowing PIM to leverage HSRP redundancy, avoid potential duplicate traffic, and enable failover, depending on the HSRP states in the device. The PIM designated router (DR) runs on the same gateway as the HSRP AR and maintains mroute states.

In a multiaccess segment (such as LAN), PIM DR election is unaware of the redundancy configuration, and the elected DR and HSRP AR may not be the same router. In order to ensure that the PIM DR is always able to forward PIM Join/Prune message towards RP or FHR, the HSRP AR becomes the PIM DR (if there is only one HSRP group). PIM is responsible for adjusting DR priority based on the group state. When a failover occurs, multicast states are created on the new AR elected by the HSRP group and the AR assumes responsibility for the routing and forwarding of all the traffic addressed to the HSRP virtual IP address.

With HSRP Aware PIM enabled, PIM sends an additional PIM Hello message using the HSRP virtual IP addresses as the source address for each active HSRP group when a device becomes HSRP Active. The PIM

Hello will carry a new GenID in order to trigger other routers to respond to the failover. When a downstream device receives this PIM Hello, it will add the virtual address to its PIM neighbor list. The new GenID carried in the PIM Hello will trigger downstream routers to resend PIM Join messages towards the virtual address. Upstream routers will process PIM Join/Prunes (J/P) based on HSRP group state.

If the J/P destination matches the HSRP group virtual address and if the destination device is in HSRP active state, the new AR processes the PIM Join because it is now the acting PIM DR. This allows all PIM Join/Prunes to reach the HSRP group virtual address and minimizes changes and configurations at the downstream routers side.

The IP routing service utilizes the existing virtual routing protocol to provide basic stateless failover services to client applications, such as PIM. Changes in the local HSRP group state and standby router responsibility are communicated to interested client applications. Client applications may build on top of IRS to provide stateful or stateless failover. PIM, as an HSRP client, listens to the state change notifications from HSRP and automatically adjusts the priority of the PIM DR based on the HSRP state. The PIM client also triggers communication between upstream and downstream devices upon failover in order to create an mroute state on the new AR.

How to Configure HSRP Aware PIM

Configuring an HSRP Group on an Interface

Before you begin

- IP multicast must already be configured on the device.
- PIM must already be configured on the interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [*secondary*]]
6. **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*
7. **standby** [*group-number*] **priority** *priority*
8. **standby** [*group-number*] **name** *group-name*
9. **end**
10. **show standby** [*type number* [*group*]] [**all** | **brief**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> [<i>name-tag</i>] Example: Device(config)# interface ethernet 0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 4	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.0.0.2 255.255.255.0	Sets a primary or secondary IP address for an interface.
Step 5	standby [<i>group-number</i>] ip [<i>ip-address</i> [<i>secondary</i>]] Example: Device(config-if)# standby 1 ip 192.0.2.99	Activates HSRP and defines an HSRP group.
Step 6	standby [<i>group-number</i>] timers [<i>msec</i>] <i>hellotime</i> [<i>msec</i>] <i>holdtime</i> Example: Device(config-if)# standby 1 timers 5 15	(Optional) Configures the time between hello packets and the time before other devices declare an HSRP active or standby router to be down.
Step 7	standby [<i>group-number</i>] priority <i>priority</i> Example: Device(config-if)# standby 1 priority 120	(Optional) Assigns the HSRP priority to be used to help select the HSRP active and standby routers.
Step 8	standby [<i>group-number</i>] name <i>group-name</i> Example: Device(config-if)# standby 1 name HSRP1	(Optional) Defines a name for the HSRP group. Note We recommend that you always configure the standby ip name command when configuring an HSRP group to be used for HSRP Aware PIM.
Step 9	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 10	show standby [<i>type number</i> [<i>group</i>]] [all brief] Example: Device# show standby	Displays HSRP group information for verifying the configuration.

Configuring PIM Redundancy

Before you begin

The HSRP group must already be configured on the interface. See the “Configuring an HSRP Group on an Interface” section.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask*
5. **ip pim redundancy group dr-priority** *priority*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> [<i>name-tag</i>] Example: Device(config)# interface ethernet 0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 4	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.0.0.2 255.255.255.0	Sets a primary or secondary IP address for an interface.
Step 5	ip pim redundancy group dr-priority <i>priority</i> Example: Device(config-if)# ip pim redundancy HSRP1 dr-priority 60	Enables PIM redundancy and assigns a redundancy priority value to the active PIM designated router (DR). <ul style="list-style-type: none"> • Because HSRP group names are case sensitive, the value of the <i>group</i> argument must match the group name configured by using the standby ip name command. • The redundancy priority for a PIM DR must be greater than the configured or default value (1) of the PIM DR priority on any device for which the same HSRP group is enabled.

	Command or Action	Purpose
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuration Examples for HSRP Aware PIM

Example: Configuring an HSRP Group on an Interface

```
interface ethernet 0/0
 ip address 10.0.0.2 255.255.255.0
 standby 1 ip 192.0.2.99
 standby 1 timers 5 15
 standby 1 priority 120
 standby 1 name HSRP1
!
```

Example: Configuring PIM Redundancy

```
interface ethernet 0/0
 ip address 10.0.0.2 255.255.255.0
 ip pim redundancy HSRP1 dr-priority 60
!
```



CHAPTER 17

Configuring VRRP Aware PIM

- [VRRP Aware PIM, on page 227](#)

VRRP Aware PIM

The Virtual Router Redundancy Protocol (VRRP) eliminates the single point of failure inherent in the static default routed environment. VRRP is an election protocol that dynamically assigns responsibility for one or more virtual routers to the VRRP routers on a LAN, allowing several routers on a multi access link to utilize the same virtual IP address.

VRRP Aware PIM is a redundancy mechanism for the Protocol Independent Multicast (PIM) to interoperate with VRRP. It allows PIM to track VRRP state and to preserve multicast traffic upon fail over in a redundant network with virtual routing groups enabled.

This module explains how to configure VRRP Aware PIM in a network.

Restrictions for VRRP Aware PIM

- Only PIM sparse mode (SM) and source specific multicast (SSM) modes are supported. Bidirectional (BiDir) PIM is not supported.
- PIM interoperability with Hot Standby Router Protocol (HSRP) IPv6 is not supported.
- PIM tracks only one virtual group, either Virtual Router Redundancy Protocol (VRRP) or HSRP, per interface.
- VRRP Aware PIM is not supported on a Transit network. PIM redundancy enabled interface does not support the PIM joining the network from down stream.

Information About VRRP Aware PIM

Overview of VRRP Aware PIM

Virtual Router Redundancy Protocol (VRRP) is a redundancy protocol for establishing a fault-tolerant default gateway. The protocol establishes a framework between network devices in order to achieve default gateway failover if the primary gateway becomes inaccessible.

Protocol Independent Multicast (PIM) has no inherent redundancy capabilities and its operation is completely independent of VRRP group states. As a result, IP multicast traffic is forwarded not necessarily by the same

device as is elected by VRRP. The VRRP Aware PIM feature provides consistent IP multicast forwarding in a redundant network with virtual routing groups enabled.

In a multi-access segment (such as LAN), PIM designated router (DR) election is unaware of the redundancy configuration, and the elected DR and VRRP primary router (MR) may not be the same router. In order to ensure that the PIM DR is always able to forward PIM Join/Prune message towards RP or FHR, the VRRP MR becomes the PIM DR (if there is only one VRRP group). PIM is responsible for adjusting DR priority based on the group state. When a fail over occurs, multicast states are created on the new MR elected by the VRRP group and the MR assumes responsibility for the routing and forwarding of all the traffic addressed to the VRRP virtual IP address. This ensures the PIM DR runs on the same gateway as the VRRP MR and maintains mroute states. It enables multicast traffic to be forwarded through the VRRP MR, allowing PIM to leverage VRRP redundancy, avoid potential duplicate traffic, and enable fail over, depending on the VRRP states in the device.

Virtual Router Redundancy Service (VRRS) provides public APIs for a client to communicate with VRRP. VRRP Aware PIM is a feature of VRRS that supports VRRPv3 (unified VRRP) in both IPv4 and IPv6.

PIM, as a VRRS client, uses the VRRS client API to obtain generic First Hop Redundancy Protocol (FHRP) state and configuration information in order to provide multicast redundancy functionalities.

PIM performs the following as a VRRS client:

- Listens to state change and update notification from VRRS server (i.e., VRRP).
- Automatically adjust PIM DR priority based on VRRP state.
- Upon VRRP fail over, PIM receives state change notification from VRRS for the tracked VRRP group and ensures traffic is forwarded through VRRP MR.

How to Configure VRRP Aware PIM

Configuring VRRP Aware PIM

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **fhrp version vrrp *version***
4. **interface *type number***
5. **ip address *address* {*primary* |*secondary*}**
6. **vrrp *group id* address-family ipv4**
7. **vrrs leader *group name***
8. **vrrp *group id* ip *ip address* {*primary* |*secondary*}**
9. **exit**
10. **interface *type number***
11. **ip pim redundancy *group name* vrrp dr-priority *priority-value***
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	fhrp version vrrp version Example: Device(config)# fhrp version vrrp v3	Enables the ability to configure VRRPv3 and VRRS.
Step 4	interface type number Example: Device(config)# interface Ethernet0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 5	ip address address {primary secondary} Example: Device(config-if)# ip address 192.0.2.2	Specifies a primary or secondary address for the VRRP group.
Step 6	vrrp group id address-family ipv4 Example: Device(config-if)# vrrp 1 address-family ipv4	Creates a VRRP group and enters VRRP configuration mode.
Step 7	vrrs leader group name Example: Device(config-if-vrrp)# vrrs leader VRRP1	Enables community and (or) extended community exchange with the specified neighbor.
Step 8	vrrp group id ip ip address {primary secondary} Example: Device(config-if-vrrp)# vrrp 1 ip 10.1.6.1	Exits address family configuration mode and returns to router configuration mode.
Step 9	exit Example: Device(config-if-vrrp)# exit	Exits VRRP configuration mode and returns to global configuration mode.

	Command or Action	Purpose
Step 10	interface <i>type number</i> Example: Device(config)# interface Ethernet0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 11	ip pim redundancy <i>group name</i> vrrp dr-priority <i>priority-value</i> Example: Device(config-if)# ip pim redundancy VRRP1 vrrp dr-priority 90	sets the priority for which a router is elected as the designated router (DR). <ul style="list-style-type: none"> The redundancy dr-priority value should be same on all routers that are enabled with VRRP Aware PIM feature.
Step 12	end Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for VRRP Aware PIM

Example: VRRP Aware PIM

```

conf terminal
  fhrp version vrrp v3
  interface Ethernet0/0
    ip address 192.0.2.2
    vrrp 1 address-family ipv4

    vrrp 1 ip 10.1.6.1

  vrrs leader VRRP1
  interface Ethernet0/0
    ip pim redundancy VRRP1 vrrp dr-priority 90
  !

```



CHAPTER 18

Configuring SSM

- [Prerequisites for Configuring SSM, on page 231](#)
- [Restrictions for Configuring SSM, on page 231](#)
- [Information About SSM and SSM Mapping, on page 233](#)
- [How to Configure SSM and SSM Mapping, on page 238](#)
- [Monitoring SSM and SSM Mapping, on page 246](#)
- [Configuration Examples for SSM and SSM Mapping, on page 247](#)

Prerequisites for Configuring SSM

The following are the prerequisites for configuring source-specific multicast (SSM) and SSM mapping:

- To use SSM and SSM mapping, you must enable IP Services feature set on 3560-CX switches.
- Before you configure SSM mapping, you must perform the following tasks:
 - Enable IP multicast routing.
 - Enable PIM sparse mode.
 - Configure SSM.
- Before you configure static SSM mapping, you must configure access control lists (ACLs) that define the group ranges to be mapped to source addresses.
- Before you can configure and use SSM mapping with DNS lookups, you need to add records to a running DNS server. If you do not already have a DNS server running, you need to install one.



Note You can use a product such as *Cisco Network Registrar* to add records to a running DNS server.

Restrictions for Configuring SSM

The following are the restrictions for configuring SSM:

- To run SSM with IGMPv3, SSM must be supported in the Cisco IOS router, the host where the application is running, and the application itself.
- Existing applications in a network predating SSM will not work within the SSM range unless they are modified to support (S, G) channel subscriptions. Therefore, enabling SSM in a network may cause problems for existing applications if they use addresses within the designated SSM range.
- IGMP Snooping—IGMPv3 uses new membership report messages that might not be correctly recognized by older IGMP snooping devices.
- Address management is still necessary to some degree when SSM is used with Layer 2 switching mechanisms. Cisco Group Management Protocol (CGMP), IGMP snooping, or Router-Port Group Management Protocol (RGMP) support only group-specific filtering, not (S, G) channel-specific filtering. If different receivers in a switched network request different (S, G) channels sharing the same group, they do not benefit from these existing mechanisms. Instead, both receivers receive all (S, G) channel traffic and filter out the unwanted traffic on input. Because SSM can re-use the group addresses in the SSM range for many independent applications, this situation can lead to decreased traffic filtering in a switched network. For this reason, it is important to use random IP addresses from the SSM range for an application to minimize the chance for re-use of a single address within the SSM range between different applications. For example, an application service providing a set of television channels should, even with SSM, use a different group for each television (S, G) channel. This setup guarantees that multiple receivers to different channels within the same application service never experience traffic aliasing in networks that include Layer 2 devices.
- In PIM-SSM, the last hop router will continue to periodically send (S, G) join messages if appropriate (S, G) subscriptions are on the interfaces. Therefore, as long as receivers send (S, G) subscriptions, the shortest path tree (SPT) state from the receivers to the source will be maintained, even if the source is not sending traffic for longer periods of time (or even never).

The opposite situation occurs with PIM-SM, where (S, G) state is maintained only if the source is sending traffic and receivers are joining the group. If a source stops sending traffic for more than 3 minutes in PIM-SM, the (S, G) state is deleted and only reestablished after packets from the source arrive again through the RPT (rendezvous point tree). Because no mechanism in PIM-SSM notifies a receiver that a source is active, the network must maintain the (S, G) state in PIM-SSM as long as receivers are requesting receipt of that channel.

The following are the restrictions for configuring SSM mapping:

- The SSM Mapping feature does not share the benefit of full SSM. SSM mapping takes a group G join from a host and identifies this group with an application associated with one or more sources, therefore, it can only support one such application per group G. Nevertheless, full SSM applications may still share the same group also used in SSM mapping.
- Enable IGMPv3 with care on the last hop router when you rely solely on SSM mapping as a transition solution for full SSM.

Information About SSM and SSM Mapping

SSM Components

SSM is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications.

SSM is a core networking technology for Cisco's implementation of IP multicast solutions targeted for audio and video broadcast application environments and is described in RFC 3569. The following components together support the implementation of SSM:

- Protocol Independent Multicast source-specific mode (PIM-SSM)
- Internet Group Management Protocol Version 3 (IGMPv3)

Protocol Independent Multicast (PIM) SSM, or PIM-SSM, is the routing protocol that supports the implementation of SSM and is derived from PIM sparse mode (PIM-SM). IGMP is the Internet Engineering Task Force (IETF) standards track protocol used for hosts to signal multicast group membership to routers. IGMP Version 3 supports source filtering, which is required for SSM. IGMP For SSM to run with IGMPv3, SSM must be supported in the router, the host where the application is running, and the application itself.

How SSM Differs from Internet Standard Multicast

The standard IP multicast infrastructure in the Internet and many enterprise intranets is based on the PIM-SM protocol and Multicast Source Discovery Protocol (MSDP). These protocols have proved to be reliable, extensive, and efficient. However, they are bound to the complexity and functionality limitations of the Internet Standard Multicast (ISM) service model. For example, with ISM, the network must maintain knowledge about which hosts in the network are actively sending multicast traffic. With SSM, this information is provided by receivers through the source addresses relayed to the last-hop devices by IGMPv3. SSM is an incremental response to the issues associated with ISM and is intended to coexist in the network with the protocols developed for ISM. In general, SSM provides IP multicast service for applications that utilize SSM.

ISM service is described in RFC 1112. This service consists of the delivery of IP datagrams from any source to a group of receivers called the multicast host group. The datagram traffic for the multicast host group consists of datagrams with an arbitrary IP unicast source address *S* and the multicast group address *G* as the IP destination address. Systems will receive this traffic by becoming members of the host group. Membership in a host group simply requires signaling the host group through IGMP Version 1, 2, or 3.

In SSM, delivery of datagrams is based on (*S, G*) channels. Traffic for one (*S, G*) channel consists of datagrams with an IP unicast source address *S* and the multicast group address *G* as the IP destination address. Systems will receive this traffic by becoming members of the (*S, G*) channel. In both SSM and ISM, no signaling is required to become a source. However, in SSM, receivers must subscribe or unsubscribe to (*S, G*) channels to receive or not receive traffic from specific sources. In other words, receivers can receive traffic only from (*S, G*) channels to which they are subscribed, whereas in ISM, receivers need not know the IP addresses of sources from which they receive their traffic. The proposed standard approach for channel subscription signaling utilizes IGMP INCLUDE mode membership reports, which are supported only in IGMP Version 3.

SSM can coexist with the ISM service by applying the SSM delivery model to a configured subset of the IP multicast group address range. The Internet Assigned Numbers Authority (IANA) has reserved the address range from 232.0.0.0 through 232.255.255.255 for SSM applications and protocols. The software allows SSM

configuration for an arbitrary subset of the IP multicast address range from 224.0.0.0 through 239.255.255.255. When an SSM range is defined, an existing IP multicast receiver application will not receive any traffic when it tries to use addresses in the SSM range unless the application is modified to use explicit (S, G) channel subscription or is SSM-enabled through a URL Rendezvous Directory (URD).

SSM Operations

An established network in which IP multicast service is based on PIM-SM can support SSM services. SSM can also be deployed alone in a network without the full range of protocols that are required for interdomain PIM-SM. That is, SSM does not require an RP, so there is no need for an RP mechanism such as Auto-RP, MSDP, or bootstrap router (BSR).

If SSM is deployed in a network that is already configured for PIM-SM, then only the last-hop routers must be upgraded to a software image that supports SSM. Routers that are not directly connected to receivers do not have to upgrade to a software image that supports SSM. In general, these non-last-hop routers must only run PIM-SM in the SSM range. They may need additional access control configuration to suppress MSDP signaling, registering, or PIM-SM shared-tree operations from occurring within the SSM range.

The SSM mode of operation is enabled by configuring the SSM range using the `ip pim ssm` global configuration command. This configuration has the following effects:

- For groups within the SSM range, (S, G) channel subscriptions are accepted through IGMPv3 INCLUDE mode membership reports.
- PIM operations within the SSM range of addresses change to PIM-SSM, a mode derived from PIM-SM. In this mode, only PIM (S, G) Join and Prune messages are generated by the router. Incoming messages related to rendezvous point tree (RPT) operations are ignored or rejected, and incoming PIM register messages are immediately answered with Register-Stop messages. PIM-SSM is backward-compatible with PIM-SM unless a router is a last-hop router. Therefore, routers that are not last-hop routers can run PIM-SM for SSM groups (for example, if they do not yet support SSM).
- For groups within the SSM range, no MSDP Source-Active (SA) messages within the SSM range will be accepted, generated, or forwarded.

IGMPv3 Host Signaling

IGMPv3 is the third version of the IETF standards track protocol in which hosts signal membership to last-hop routers of multicast groups. IGMPv3 introduces the ability for hosts to signal group membership that allows filtering capabilities with respect to sources. A host can signal either that it wants to receive traffic from all sources sending to a group except for some specific sources (a mode called EXCLUDE) or that it wants to receive traffic only from some specific sources sending to the group (a mode called INCLUDE).

IGMPv3 can operate with both ISM and SSM. In ISM, both EXCLUDE and INCLUDE mode reports are accepted by the last-hop router. In SSM, only INCLUDE mode reports are accepted by the last-hop router.

Benefits of

IP Multicast Address Management Not Required

In the ISM service, applications must acquire a unique IP multicast group address because traffic distribution is based only on the IP multicast group address used. If two applications with different sources and receivers

use the same IP multicast group address, then receivers of both applications will receive traffic from the senders of both applications. Even though the receivers, if programmed appropriately, can filter out the unwanted traffic, this situation would cause generally unacceptable levels of unwanted traffic.

Allocating a unique IP multicast group address for an application is still a problem. Most short-lived applications use mechanisms like Session Description Protocol (SDP) and Session Announcement Protocol (SAP) to get a random address, a solution that does not work well with a rising number of applications in the Internet. The best current solution for long-lived applications is described in RFC 2770, but this solution suffers from the restriction that each autonomous system is limited to only 255 usable IP multicast addresses.

In SSM, traffic from each source is forwarded between routers in the network independent of traffic from other sources. Thus different sources can reuse multicast group addresses in the SSM range.

Denial of Service Attacks from Unwanted Sources Inhibited

In SSM, multicast traffic from each individual source will be transported across the network only if it was requested (through IGMPv3, IGMP v3lite, or URD memberships) from a receiver. In contrast, ISM forwards traffic from any active source sending to a multicast group to all receivers requesting that multicast group. In Internet broadcast applications, this ISM behavior is highly undesirable because it allows unwanted sources to easily disturb the actual Internet broadcast source by simply sending traffic to the same multicast group. This situation depletes bandwidth at the receiver side with unwanted traffic and thus disrupts the undisturbed reception of the Internet broadcast. In SSM, this type of denial of service (DoS) attack cannot be made by simply sending traffic to a multicast group.

Easy to Install and Manage

SSM is easy to install and provision in a network because it does not require the network to maintain which active sources are sending to multicast groups. This requirement exists in ISM (with IGMPv1, IGMPv2, or IGMPv3).

The current standard solutions for ISM service are PIM-SM and MSDP. Rendezvous point (RP) management in PIM-SM (including the necessity for Auto-RP or BSR) and MSDP is required only for the network to learn about active sources. This management is not necessary in SSM, which makes SSM easier than ISM to install and manage, and therefore easier than ISM to operationally scale in deployment. Another factor that contributes to the ease of installation of SSM is the fact that it can leverage preexisting PIM-SM networks and requires only the upgrade of last hop routers to support IGMPv3, IGMP v3lite, or URD.

Ideal for Internet Broadcast Applications

The three benefits previously described make SSM ideal for Internet broadcast-style applications for the following reasons:

- The ability to provide Internet broadcast services through SSM without the need for unique IP multicast addresses allows content providers to easily offer their service (IP multicast address allocation has been a serious problem for content providers in the past).
- The prevention against DoS attacks is an important factor for Internet broadcast services because, with their exposure to a large number of receivers, they are the most common targets for such attacks.
- The ease of installation and operation of SSM makes it ideal for network operators, especially in those cases where content needs to be forwarded between multiple independent PIM domains (because there is no need to manage MSDP for SSM between PIM domains).

SSM Mapping Overview

SSM mapping supports SSM transition when supporting SSM on the end system is impossible or unwanted due to administrative or technical reasons. Using SSM to deliver live streaming video to legacy STBs that do not support IGMPv3 is a typical application of SSM mapping.

In a typical STB deployment, each TV channel uses one separate IP multicast group and has one active server host sending the TV channel. A single server may of course send multiple TV channels, but each to a different group. In this network environment, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group G, the report implicitly addresses the well-known TV server for the TV channel associated with the multicast group.

SSM mapping introduces a means for the last hop router to discover sources sending to groups. When SSM mapping is configured, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group G, the router translates this report into one or more (S, G) channel memberships for the well-known sources associated with this group.

When the router receives an IGMPv1 or IGMPv2 membership report for group G, the router uses SSM mapping to determine one or more source IP addresses for group G. SSM mapping then translates the membership report as an IGMPv3 report INCLUDE (G, [S1, G], [S2, G]...[Sn, G]) and continues as if it had received an IGMPv3 report. The router then sends out PIM joins toward (S1, G) to (Sn, G) and continues to be joined to these groups as long as it continues to receive the IGMPv1 or IGMPv2 membership reports and as long as the SSM mapping for the group remains the same. SSM mapping, thus, enables you to leverage SSM for video delivery to legacy STBs that do not support IGMPv3 or for applications that do not take advantage of the IGMPv3 host stack.

SSM mapping enables the last hop router to determine the source addresses either by a statically configured table on the router or by consulting a DNS server. When the statically configured table is changed, or when the DNS mapping changes, the router will leave the current sources associated with the joined groups.

Static SSM Mapping

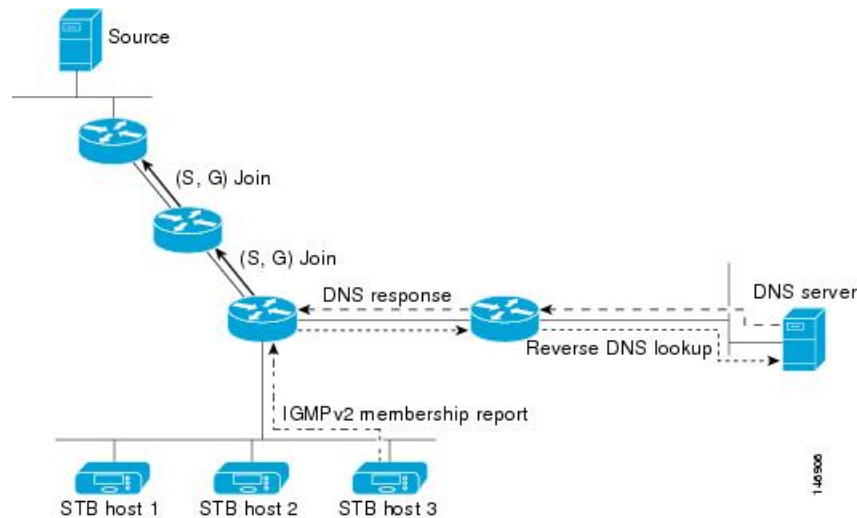
SSM static mapping enables you to configure the last hop router to use a static map to determine the sources sending to groups. Static SSM mapping requires that you configure access lists (ACLs) to define group ranges. The groups permitted by those ACLs then can be mapped to sources using the **ip igmp static ssm-map** global configuration command.

You can configure static SSM mapping in smaller networks when a DNS is not needed or to locally override DNS mappings that may be temporarily incorrect. When configured, static SSM mappings take precedence over DNS mappings.

DNS-Based SSM Mapping

DNS-based SSM mapping enables you to configure the last hop router to perform a reverse DNS lookup to determine sources sending to groups (see the figure below). When DNS-based SSM mapping is configured, the router constructs a domain name that includes the group address G and performs a reverse lookup into the DNS. The router looks up IP address resource records (IP A RRs) to be returned for this constructed domain name and uses the returned IP addresses as the source addresses associated with this group. SSM mapping supports up to 20 sources for each group. The router joins all sources configured for a group.

Figure 11: DNS-Based SSM-Mapping



The SSM mapping mechanism that enables the last hop router to join multiple sources for a group can be used to provide source redundancy for a TV broadcast. In this context, the redundancy is provided by the last hop router using SSM mapping to join two video sources simultaneously for the same TV channel. However, to prevent the last hop router from duplicating the video traffic, it is necessary that the video sources utilize a server-side switchover mechanism where one video source is active while the other backup video source is passive. The passive source waits until an active source failure is detected before sending the video traffic for the TV channel. The server-side switchover mechanism, thus, ensures that only one of the servers is actively sending the video traffic for the TV channel.

To look up one or more source addresses for a group G that includes G1, G2, G3, and G4, the following DNS resource records (RRs) must be configured on the DNS server:

G4.G3.G2.G1 [<i>multicast-domain</i>] [<i>timeout</i>]	IN A <i>source-address-1</i>
	IN A <i>source-address-2</i>
	IN A <i>source-address-n</i>

The *multicast-domain* argument is a configurable DNS prefix. The default DNS prefix is `in-addr.arpa`. You should only use the default prefix when your installation is either separate from the internet or if the group names that you map are global scope group addresses (RFC 2770 type addresses that you configure for SSM) that you own.

The *timeout* argument configures the length of time for which the router performing SSM mapping will cache the DNS lookup. This argument is optional and defaults to the timeout of the zone in which this entry is configured. The timeout indicates how long the router will keep the current mapping before querying the DNS server for this group. The timeout is derived from the cache time of the DNS RR entry and can be configured for each group/source entry on the DNS server. You can configure this time for larger values if you want to minimize the number of DNS queries generated by the router. Configure this time for a low value if you want to be able to quickly update all routers with new source addresses.



Note Refer to your DNS server documentation for more information about configuring DNS RRs.

To configure DNS-based SSM mapping in the software, you must configure a few global commands but no per-channel specific configuration is needed. There is no change to the configuration for SSM mapping if additional channels are added. When DNS-based SSM mapping is configured, the mappings are handled entirely by one or more DNS servers. All DNS techniques for configuration and redundancy management can be applied to the entries needed for DNS-based SSM mapping.

SSM Mapping Benefits

- The SSM Mapping feature provides almost the same ease of network installation and management as a pure SSM solution based on IGMPv3. Some additional configuration is necessary to enable SSM mapping.
- The SSM benefit of inhibition of DoS attacks applies when SSM mapping is configured. When SSM mapping is configured the only segment of the network that may still be vulnerable to DoS attacks are receivers on the LAN connected to the last hop router. Since those receivers may still be using IGMPv1 and IGMPv2, they are vulnerable to attacks from unwanted sources on the same LAN. SSM mapping, however, does protect those receivers (and the network path leading towards them) from multicast traffic from unwanted sources anywhere else in the network.
- Address assignment within a network using SSM mapping needs to be coordinated, but it does not need assignment from outside authorities, even if the content from the network is to be transited into other networks.

How to Configure SSM and SSM Mapping

Configuring SSM

Follow these steps to configure SSM:

This procedure is optional.

Before you begin

If you want to use an access list to define the Source Specific Multicast (SSM) range, configure the access list before you reference the access list in the **ip pim ssm** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip pim ssm [default | range *access-list*]**
4. **interface *type number***
5. **ip pim {sparse-mode | sparse-dense-mode}**
6. **ip igmp version 3**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip pim ssm [default range <i>access-list</i>] Example: <pre>Switch(config)# ip pim ssm range 20</pre>	Defines the SSM range of IP multicast addresses.
Step 4	interface <i>type number</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Selects an interface that is connected to hosts on which IGMPv3 can be enabled, and enters the interface configuration mode.
Step 5	ip pim {sparse-mode sparse-dense-mode} Example: <pre>Switch(config-if)# ip pim sparse-mode</pre>	Enables PIM on an interface.
Step 6	ip igmp version 3 Example: <pre>Switch(config-if)# ip igmp version 3</pre>	Enables IGMPv3 on this interface. The default version of IGMP is set to Version 2.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show running-config</code>	
Step 9	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring SSM Mapping

Configuring Static SSM Mapping

Follow these steps to configure static SSM Mapping:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip igmp ssm-map enable`
4. `no ip igmp ssm-map query dns`
5. `ip igmp ssm-map static access-list source-address`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip igmp ssm-map enable Example: Switch(config)# <code>ip igmp ssm-map enable</code>	Enables SSM mapping for groups in the configured SSM range. <p>Note By default, this command enables DNS-based SSM mapping.</p>

	Command or Action	Purpose
Step 4	<p>no ip igmp ssm-map query dns</p> <p>Example:</p> <pre>Switch(config)# no ip igmp ssm-map query dns</pre>	<p>(Optional) Disables DNS-based SSM mapping.</p> <p>Note Disable DNS-based SSM mapping if you only want to rely on static SSM mapping. By default, the ip igmp ssm-map command enables DNS-based SSM mapping.</p>
Step 5	<p>ip igmp ssm-map static <i>access-list source-address</i></p> <p>Example:</p> <pre>Switch(config)# ip igmp ssm-map static 11 172.16.8.11</pre>	<p>Configures static SSM mapping.</p> <ul style="list-style-type: none"> The ACL supplied for the <i>access-list</i> argument defines the groups to be mapped to the source IP address entered for the <i>source-address</i> argument. <p>Note You can configure additional static SSM mappings. If additional SSM mappings are configured and the router receives an IGMPv1 or IGMPv2 membership report for a group in the SSM range, the device determines the source addresses associated with the group by walking each configured ip igmp ssm-map static command. The device associates up to 20 sources per group.</p> <p>Repeat Step to configure additional static SSM mappings, if required.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Configuring DNS-Based SSM Mapping (CLI)

Perform this task to configure the last hop router to perform DNS lookups to learn the IP addresses of sources sending to a group.

Before you begin

- Enable IP multicast routing, enable PIM sparse mode, and configure SSM before performing this task. For more information, see the "Configuring Basic Multicast" module.
- Before you can configure and use SSM mapping with DNS lookups, you need to be able to add records to a running DNS server. If you do not already have a DNS server running, you need to install one.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp ssm-map enable**
4. **ip igmp ssm-map query dns**
5. **ip domain multicast** *domain-prefix*
6. **ipname-server** *server-address1* [*server-address2server-address6*]
7. Repeat Step 6 to configure additional DNS servers for redundancy, if required.
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device# enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp ssm-map enable Example: Device(config)# ip igmp ssm-map enable	Enables SSM mapping for groups in a configured SSM range.
Step 4	ip igmp ssm-map query dns Example: Device(config)# ip igmp ssm-map query dns	(Optional) Enables DNS-based SSM mapping. • By default, the ip igmp ssm-map command enables DNS-based SSM mapping. Only the no form of this command is saved to the running configuration. Note Use this command to reenable DNS-based SSM mapping if DNS-based SSM mapping is disabled.
Step 5	ip domain multicast <i>domain-prefix</i> Example:	(Optional) Changes the domain prefix used by the Cisco IOS XE software for DNS-based SSM mapping.

	Command or Action	Purpose
	Device(config)# ip domain multicast ssm-map.cisco.com	<ul style="list-style-type: none"> By default, the software uses the ip-addr.arpa domain prefix.
Step 6	ipname-server <i>server-address1</i> [<i>server-address2server-address6</i>] Example: Device(config)# ip name-server 10.48.81.21	Specifies the address of one or more name servers to use for name and address resolution.
Step 7	Repeat Step 6 to configure additional DNS servers for redundancy, if required.	--
Step 8	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Static Traffic Forwarding with SSM Mapping

Follow these steps to configure static traffic forwarding with SSM mapping on the last hop router:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp static-group** *group-address* **source ssm-map**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Selects an interface on which to statically forward traffic for a multicast group using SSM mapping, and enters interface configuration mode. Note Static forwarding of traffic with SSM mapping works with either DNS-based SSM mapping or statically configured SSM mapping.
Step 4	ip igmp static-group <i>group-address</i> source ssm-map Example: Switch(config-if)# ip igmp static-group 239.1.2.1 source ssm-map	Configures SSM mapping to statically forward a (S, G) channel from the interface. Use this command if you want to statically forward SSM traffic for certain groups. Use DNS-based SSM mapping to determine the source addresses of the channels.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Verifying SSM Mapping Configuration and Operation

Follow these steps to verify SSM mapping configuration and operation:

SUMMARY STEPS

1. `enable`
2. `show ip igmp ssm-mapping`
3. `show ip igmp ssm-mapping group-address`
4. `show ip igmp groups [group-name | group-address | interface-type interface-number] [detail]`
5. `show host`
6. `debug ip igmp group-address`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>show ip igmp ssm-mapping</code></p> <p>Example:</p> <pre>Switch# show ip igmp ssm-mapping SSM Mapping : Enabled DNS Lookup : Enabled Mcast domain : ssm-map.cisco.com Name servers : 10.0.0.3 10.0.0.4</pre>	<p>(Optional) Displays information about SSM mapping configuration.</p>
Step 3	<p><code>show ip igmp ssm-mapping group-address</code></p> <p>Example:</p> <pre>Switch# show ip igmp ssm-mapping 232.1.1.4 Group address: 232.1.1.4 Database : DNS DNS name : 4.1.1.232.ssm-map.cisco.com Expire time : 860000 Source list : 172.16.8.5 : 172.16.8.6</pre>	<p>(Optional) Displays the sources that SSM mapping uses for a particular group.</p> <p>The example here shows information about the configured DNS-based SSM mapping. Here the router has used DNS-based mapping to map group 232.1.1.4 to sources 172.16.8.5 and 172.16.8.6. The timeout for this entry is 860000 milliseconds (860 seconds).</p>
Step 4	<p><code>show ip igmp groups [group-name group-address interface-type interface-number] [detail]</code></p> <p>Example:</p> <pre>Switch# show ip igmp group 232.1.1.4 detail Interface: GigabitEthernet2/0/0 Group: 232.1.1.4 SSM Uptime: 00:03:20 Group mode: INCLUDE Last reporter: 0.0.0.0 CSR Grp Exp: 00:02:59 Group source list: (C - Cisco Src Report, U - URD, R - Remote, S - Static, M - SSM Mapping)</pre>	<p>(Optional) Displays the multicast groups with receivers that are directly connected to the router and that were learned through IGMP.</p> <p>In the example the “M” flag indicates that SSM mapping is configured.</p>

	Command or Action	Purpose
	<pre> Source Address Uptime v3 Exp CSR Exp Fwd Flags 172.16.8.3 00:03:20 stopped 00:02:59 Yes CM 172.16.8.4 00:03:20 stopped 00:02:59 Yes CM 172.16.8.5 00:03:20 stopped 00:02:59 Yes CM 172.16.8.6 00:03:20 stopped 00:02:59 Yes CM </pre>	
Step 5	<p>show host</p> <p>Example:</p> <pre> Switch# show host Default domain is cisco.com Name/address lookup uses domain service Name servers are 10.48.81.21 Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate temp - temporary, perm - permanent NA - Not Applicable None - Not defined Host Port Flags Age Type Address(es) 10.0.0.0.ssm-map.cisco.c None (temp, OK) 0 IP 172.16.8.5 172.16.8.6 172.16.8.3 </pre>	(Optional) Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.
Step 6	<p>debug ip igmp group-address</p> <p>Example:</p> <pre> Switch# debug ip igmp IGMP(0): Convert IGMPv2 report (*,232.1.2.3) to IGMPv3 with 2 source(s) using STATIC. Switch# debug ip igmp IGMP(0): Convert IGMPv2 report (*,232.1.2.3) to IGMPv3 with 2 source(s) using DNS. Switch# debug ip igmp IGMP(0): DNS source lookup failed for (*, 232.1.2.3), IGMPv2 report failed </pre>	<p>(Optional) Displays the IGMP packets received and sent and IGMP host-related events.</p> <p>In the first example, the output indicates that the router is converting an IGMPv2 join for group G into an IGMPv3 join.</p> <p>In the second example, the output indicates that a DNS lookup has succeeded.</p> <p>In the third example, the output indicates that DNS-based SSM mapping is enabled and a DNS lookup has failed:</p>

Monitoring SSM and SSM Mapping

Monitoring SSM

To monitor SSM, use the following commands in privileged EXEC mode, as needed:

Command	Purpose
Switch# show ip igmp groups detail	Displays the (S, G) channel subscription through IGMPv3.
Switch# show ip mroute	Displays whether a multicast group supports SSM service or whether a source-specific host report was received.

Monitoring SSM Mapping

Use the privileged EXEC commands in the following table to monitor SSM mapping.

Table 31: SSM Mapping Monitoring Commands

Command	Purpose
Switch# show ip igmp ssm-mapping	Displays information about SSM mapping.
Switch# show ip igmp ssm-mapping group-address	Displays the sources that SSM mapping uses group.
Switch# show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>interface-type interface-number</i>] [detail]	Displays the multicast groups with receivers connected to the router and that were learned
Switch# show host	Displays the default domain name, the style of service, a list of name server hosts, and the ca hostnames and addresses.
Switch# debug ip igmp group-address	Displays the IGMP packets received and sent host-related events.

Configuration Examples for SSM and SSM Mapping

SSM with IGMPv3 Example

The following example shows how to configure a device (running IGMPv3) for SSM:

```
ip multicast-routing
!
interface GigabitEthernet3/1/0
 ip address 172.21.200.203 255.255.255.0
 description backbone interface
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0
 ip address 131.108.1.2 255.255.255.0
 ip pim sparse-mode
 description ethernet connected to hosts
 ip igmp version 3
```

```
!
ip pim ssm default
```

SSM Filtering Example

The following example shows how to configure filtering on legacy RP routers running software releases that do not support SSM routing. This filtering will suppress all unwanted PIM-SM and MSDP traffic in the SSM range. Without this filtering, SSM will still operate, but there may be additional RPT traffic if legacy first hop and last hop routers exist in the network.

```
ip access-list extended no-ssm-range
  deny ip any 232.0.0.0 0.255.255.255 ! SSM range
  permit ip any any
! Deny registering in SSM range
ip pim accept-register list no-ssm-range
ip access-list extended msdp-nono-list
  deny ip any 232.0.0.0 0.255.255.255 ! SSM Range
  ! .
  ! .
  ! .
! See ftp://ftpeng.cisco.com/ipmulticast/config-notes/msdp-sa-filter.txt for other SA
! messages that typically need to be filtered.
  permit ip any any
! Filter generated SA messages in SSM range. This configuration is only needed if there
! are directly connected sources to this router. The "ip pim accept-register" command
! filters remote sources.
ip msdp redistribute list msdp-nono-list
! Filter received SA messages in SSM range. "Filtered on receipt" means messages are
! neither processed or forwarded. Needs to be configured for each MSDP peer.
ip msdp sa-filter in msdp-peer1 list msdp-nono-list
! .
! .
! .
ip msdp sa-filter in msdp-peerN list msdp-nono-list
```

SSM Mapping Example

The following configuration example shows a router configuration for SSM mapping. This example also displays a range of other IGMP and SSM configuration options to show compatibility between features. Do not use this configuration example as a model unless you understand all of the features used in the example.



Note Address assignment in the global SSM range 232.0.0.0/8 should be random. If you copy parts or all of this sample configuration, make sure to select a random address range but not 232.1.1.x as shown in this example. Using a random address range minimizes the possibility of address collision and may prevent conflicts when other SSM content is imported while SSM mapping is used.

```
!
no ip domain lookup
ip domain multicast ssm.map.cisco.com
ip name-server 10.48.81.21
!
!
ip multicast-routing distributed
ip igmp ssm-map enable
```



```

ip igmp ssm-map static 10 172.16.8.10
ip igmp ssm-map static 11 172.16.8.11
!
!
.
.
.
!
interface GigabitEthernet0/0/0
description Sample IGMP Interface Configuration for SSM-Mapping Example
ip address 10.20.1.2 255.0.0.0
ip pim sparse-mode
ip igmp last-member-query-interval 100
ip igmp static-group 232.1.2.1 source ssm-map
ip igmp version 3
ip igmp explicit-tracking
ip igmp limit 2
ip igmp v3lite
ip urd
!
.
.
.
!
ip pim ssm default
!
access-list 10 permit 232.1.2.10
access-list 11 permit 232.1.2.0 0.0.0.255
!

```

This table describes the significant commands shown in the SSM mapping configuration example.

Table 32: SSM Mapping Configuration Example Command Descriptions

Command	Description
no ip domain lookup	Disables IP DNS-based hostname-to-address translation. Note The no ip domain-list command is shown in the configuration only to demonstrate that disabling IP DNS-based hostname-to-address translation does not conflict with configuring SSM mapping. If this command is enabled, the Cisco IOS XE software will try to resolve unknown strings as hostnames.
ip domain multicast ssm-map.cisco.com	Specifies ssm-map.cisco.com as the domain prefix for SSM mapping.
ip name-server 10.48.81.21	Specifies 10.48.81.21 as the IP address of the DNS server to be used by SSM mapping and any other service in the software that utilizes DNS.
ip multicast-routing	Enables IP multicast routing.
ip igmp ssm-map enable	Enables SSM mapping.
ip igmp ssm-map static 10 172.16.8.10	Configures the groups permitted by ACL 10 to use source address 172.16.8.10. <ul style="list-style-type: none"> In this example, ACL 10 permits all groups in the 232.1.2.0/25 range except 232.1.2.10.

Command	Description
ip igmp ssm-map static 11 172.16.8.11	Configures the groups permitted by ACL 11 to use source address 172.16.8.11. • In this example, ACL 11 permits group 232.1.2.10.
ip pim sparse-mode	Enables PIM sparse mode.
ip igmp last-member-query-interval 100	Reduces the leave latency for IGMPv2 hosts. Note This command is not required for configuring SSM mapping; however, configuring this command can be beneficial for IGMPv2 hosts relying on SSM mapping.
ip igmp static-group 232.1.2.1 source ssm-map	Configures SSM mapping to be used to determine the sources associated with group 232.1.2.1. The resulting (S, G) channels are statically forwarded.
ip igmp version 3	Enables IGMPv3 on this interface. Note This command is shown in the configuration only to demonstrate that IGMPv3 can be configured simultaneously with SSM mapping; however, it is not required.
ip igmp explicit-tracking	Minimizes the leave latency for IGMPv3 host leaving a multicast channel. Note This command is not required for configuring SSM mapping.
ip igmp limit 2	Limits the number of IGMP states resulting from IGMP membership states on a per-interface basis. Note This command is not required for configuring SSM mapping.
ip igmp v3lite	Enables the acceptance and processing of IGMP v3lite membership reports on this interface. Note This command is shown in the configuration only to demonstrate that IGMP v3lite can be configured simultaneously with SSM mapping; however, it is not required.
ip urd	Enables interception of TCP packets sent to the reserved URD port 465 on an interface and processing of URD channel subscription reports. Note This command is shown in the configuration only to demonstrate that URD can be configured simultaneously with SSM mapping; however, it is not required.
ip pim ssm default	Configures SSM service. The default keyword defines the SSM range access list as 232/8.

Command	Description
<pre>access-list 10 permit 232.1.2.10 access-list 11 permit 232.1.2.0 0.0.0.255</pre>	<p>Configures the ACLs to be used for static SSM mapping.</p> <p>Note These are the ACLs that are referenced by the ip igmp ssm-map static commands in this configuration example.</p>

DNS Server Configuration Example

To configure DNS-based SSM mapping, you need to create a DNS server zone or add records to an existing zone. If the routers that are using DNS-based SSM mapping are also using DNS for other purposes besides SSM mapping, you should use a normally-configured DNS server. If DNS-based SSM mapping is the only DNS implementation being used on the router, you can configure a fake DNS setup with an empty root zone, or a root zone that points back to itself.

The following example shows how to create a zone and import the zone data using Network Registrar:

```
Router> zone 1.1.232.ssm-map.cisco.com. create primary file=named.ssm-map
100 Ok
Router> dns reload
100 Ok
```

The following example shows how to import the zone files from a named.conf file for BIND 8:

```
Router> ::import named.conf /etc/named.conf
Router> dns reload
100 Ok:
```



Note Network Registrar version 8.0 and later support import BIND 8 format definitions.



CHAPTER 19

Configuring MSDP

- [Prerequisites for MSDP, on page 253](#)
- [Information About Multicast Source Discovery Protocol, on page 253](#)
- [How to Configure MSDP, on page 260](#)
- [Monitoring and Maintaining MSDP, on page 278](#)
- [Configuration Examples for Configuring MSDP, on page 282](#)

Prerequisites for MSDP

To use MSDP, you must enable IP services feature set on Catalyst 3560-CX switches.

Information About Multicast Source Discovery Protocol

MSDP is a mechanism to connect multiple PIM-SM domains. The purpose of MSDP is to discover multicast sources in other PIM domains. The main advantage of MSDP is that it reduces the complexity of interconnecting multiple PIM-SM domains by allowing PIM-SM domains to use an interdomain source tree (rather than a common shared tree). When MSDP is configured in a network, RPs exchange source information with RPs in other domains. An RP can join the interdomain source tree for sources that are sending to groups for which it has receivers. The RP can do that because it is the root of the shared tree within its domain, which has branches to all points in the domain where there are active receivers. When a last-hop device learns of a new source outside the PIM-SM domain (through the arrival of a multicast packet from the source down the shared tree), it then can send a join toward the source and join the interdomain source tree.



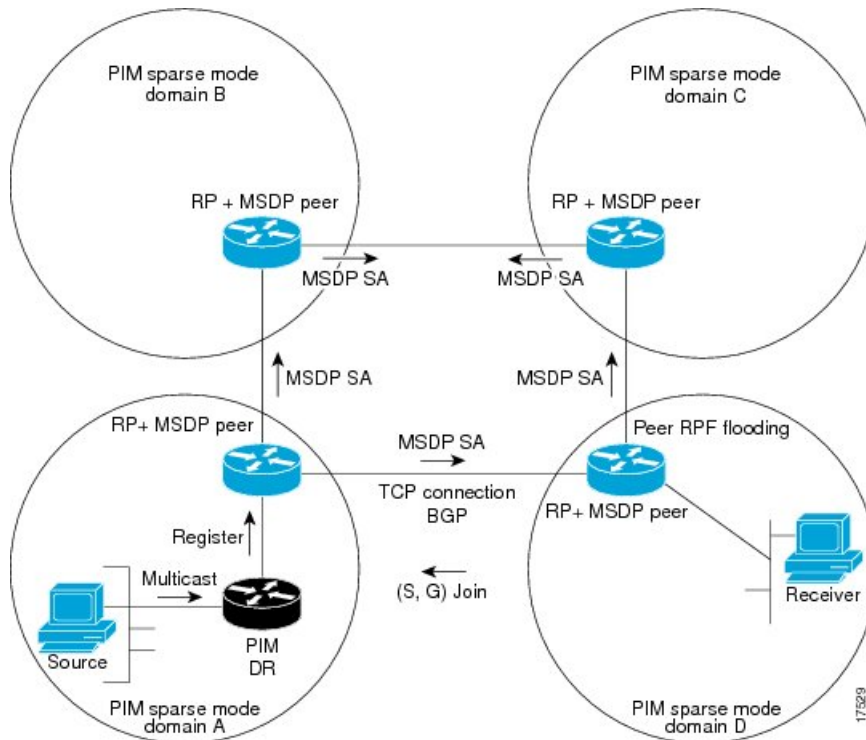
Note If the RP either has no shared tree for a particular group or a shared tree whose outgoing interface list is null, it does not send a join to the source in another domain.

When MSDP is enabled, an RP in a PIM-SM domain maintains MSDP peering relationships with MSDP-enabled devices in other domains. This peering relationship occurs over a TCP connection, where primarily a list of sources sending to multicast groups is exchanged. MSDP uses TCP (port 639) for its peering connections. As with BGP, using point-to-point TCP peering means that each peer must be explicitly configured. The TCP connections between RPs, moreover, are achieved by the underlying routing system. The receiving RP uses the source lists to establish a source path. If the multicast sources are of interest to a domain that has

receivers, multicast data is delivered over the normal, source-tree building mechanism provided by PIM-SM. MSDP is also used to announce sources sending to a group. These announcements must originate at the RP of the domain.

The figure illustrates MSDP operating between two MSDP peers. PIM uses MSDP as the standard mechanism to register a source with the RP of a domain.

Figure 12: MSDP Running Between RP Peers



When MSDP is implemented, the following sequence of events occurs:

1. When a PIM designated device (DR) registers a source with its RP as illustrated in the figure, the RP sends a Source-Active (SA) message to all of its MSDP peers.



Note The DR sends the encapsulated data to the RP only once per source (when the source goes active). If the source times out, this process happens again when it goes active again. This situation is different from the periodic SA message that contains all sources that are registered to the originating RP. Those SA messages are MSDP control packets, and, thus, do not contain encapsulated data from active sources.

1. The SA message identifies the source address, the group that the source is sending to, and the address or the originator ID of the RP, if configured.
2. Each MSDP peer that receives the SA message floods the SA message to all of its peers downstream from the originator. In some cases (such as the case with the RPs in PIM-SM domains B and C in the figure), an RP may receive a copy of an SA message from more than one MSDP peer. To prevent looping, the RP consults the BGP next-hop database to determine the next hop toward the originator of the SA message. If both MBGP and unicast BGP are configured, MBGP is checked first, and then unicast BGP. That

next-hop neighbor is the RPF-peer for the originator. SA messages that are received from the originator on any interface other than the interface to the RPF peer are dropped. The SA message flooding process, therefore, is referred to as peer-RPF flooding. Because of the peer-RPF flooding mechanism, BGP or MBGP must be running in conjunction with MSDP.

1. When an RP receives an SA message, it checks to see whether there are any members of the advertised groups in its domain by checking to see whether there are interfaces on the group's (*, G) outgoing interface list. If there are no group members, the RP does nothing. If there are group members, the RP sends an (S, G) join toward the source. As a result, a branch of the interdomain source tree is constructed across autonomous system boundaries to the RP. As multicast packets arrive at the RP, they are then forwarded down its own shared tree to the group members in the RP's domain. The members' DRs then have the option of joining the rendezvous point tree (RPT) to the source using standard PIM-SM procedures.
2. The originating RP continues to send periodic SA messages for the (S, G) state every 60 seconds for as long as the source is sending packets to the group. When an RP receives an SA message, it caches the SA message. Suppose, for example, that an RP receives an SA message for (172.16.5.4, 228.1.2.3) from originating RP 10.5.4.3. The RP consults its mroute table and finds that there are no active members for group 228.1.2.3, so it passes the SA message to its peers downstream of 10.5.4.3. If a host in the domain then sends a join to the RP for group 228.1.2.3, the RP adds the interface toward the host to the outgoing interface list of its (*, 224.1.2.3) entry. Because the RP caches SA messages, the device will have an entry for (172.16.5.4, 228.1.2.3) and can join the source tree as soon as a host requests a join.



Note In all current and supported software releases, caching of MSDP SA messages is mandatory and cannot be manually enabled or disabled. By default, when an MSDP peer is configured, the **ip multicast cache-sa-state** command will automatically be added to the running configuration.

MSDP Benefits

MSDP has these benefits:

- It breaks up the shared multicast distribution tree. You can make the shared tree local to your domain. Your local members join the local tree, and join messages for the shared tree never need to leave your domain.
- PIM sparse-mode domains can rely only on their own RPs, decreasing reliance on RPs in another domain. This increases security because you can prevent your sources from being known outside your domain.
- Domains with only receivers can receive data without globally advertising group membership.
- Global source multicast routing table state is not required, saving memory.

Default MSDP Peers

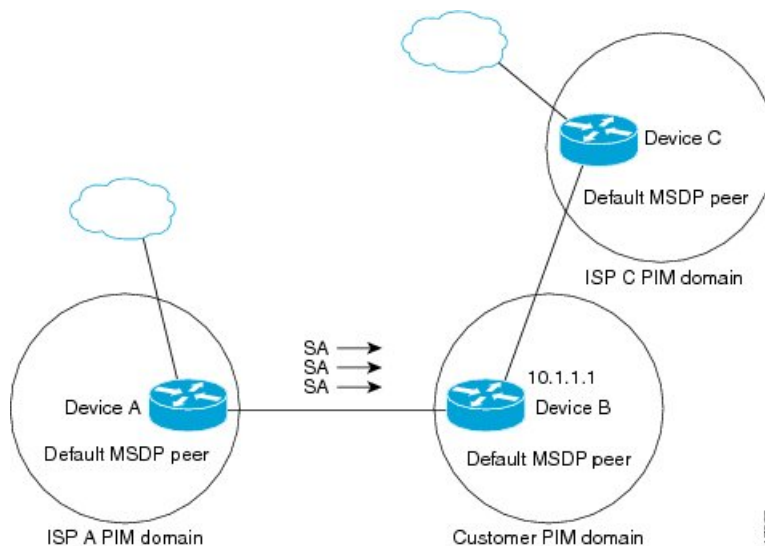
A stub autonomous system also might want to have MSDP peerings with more than one RP for the sake of redundancy. For example, SA messages cannot just be accepted from multiple default peers, because there is no RPF check mechanism. Instead, SA messages are accepted from only one peer. If that peer fails, SA messages are then accepted from the other peer. The underlying assumption here, of course, is that both default peers are sending the same SA messages.

The figure illustrates a scenario where default MSDP peers might be used. In the figure, a customer that owns Device B is connected to the Internet through two Internet service providers (ISPs), one that owns Device A and the other that owns Device C. They are not running BGP or MBGP between them. In order for the customer to learn about sources in the ISP domain or in other domains, Device B identifies Device A as its default MSDP peer. Device B advertises SA messages to both Device A and Device C, but accepts SA messages either from Device A only or Device C only. If Device A is the first default peer in the configuration, it will be used if it is up and running. Only if Device A is not running will Device B accept SA messages from Device C.

The ISP will also likely use a prefix list to define which prefixes it will accept from the customer device. The customer will define multiple default peers, each having one or more prefixes associated with it.

The customer has two ISPs to use. The customer defines both ISPs as default peers. As long as the first default peer identified in the configuration is up and running, it will be the default peer and the customer will accept all SA messages it receives from that peer.

Figure 13: Default MSDP Peer Scenario



Device B advertises SAs to Device A and Device C, but uses only Device A or Device C to accept SA messages. If Device A is first in the configuration, it will be used if it is up and running. Only when Device A is not running will Device B accept SAs from Device C. This is the behavior without a prefix list.

If you specify a prefix list, the peer will be a default peer only for the prefixes in the list. You can have multiple active default peers when you have a prefix list associated with each. When you do not have any prefix lists, you can configure multiple default peers, but only the first one is the active default peer as long as the device has connectivity to this peer and the peer is alive. If the first configured peer goes down or the connectivity to this peer goes down, the second configured peer becomes the active default, and so on.

MSDP Mesh Groups

An MSDP mesh group is a group of MSDP speakers that have fully meshed MSDP connectivity between one another. In other words, each of the MSDP peers in the group must have an MSDP peering relationship (MSDP connection) to every other MSDP peer in the group. When an MSDP mesh group is configured between a group of MSDP peers, SA message flooding is reduced. Because when an MSDP peer in the group receives an SA message from another MSDP peer in the group, it assumes that this SA message was sent to all the

other MSDP peers in the group. As a result, it is not necessary for the receiving MSDP peer to flood the SA message to the other MSDP peers in the group.

Benefits of MSDP Mesh Groups

- Optimizes SA flooding--MSDP mesh groups are particularly useful for optimizing SA flooding when two or more peers are in a group.
- Reduces the amount of SA traffic across the Internet--When MSDP mesh groups are used, SA messages are not flooded to other mesh group peers.
- Eliminates RPF checks on arriving SA messages--When an MSDP mesh group is configured, SA messages are always accepted from mesh group peers.

SA Origination Filters

By default, an RP that is configured to run MSDP will originate SA messages for all local sources for which it is the RP. Local sources that register with an RP, therefore, will be advertised in SA messages, which in some cases is not desirable. For example, if sources inside a PIM-SM domain are using private addresses (for example, network 10.0.0.0/8), you should configure an SA origination filter to restrict those addresses from being advertised to other MSDP peers across the Internet.

To control what sources are advertised in SA messages, you can configure SA origination filters on an RP. By creating SA origination filters, you can control the sources advertised in SA messages as follows:

- You can configure an RP to prevent the device from advertising local sources in SA messages. The device will still forward SA messages from other MSDP peers in the normal fashion; it will just not originate any SA messages for local sources.
- You can configure the device to only originate SA messages for local sources sending to specific groups that match (S, G) pairs defined in the extended access list. All other local sources will not be advertised in SA messages.
- You can configure the device to only originate SA messages for local sources sending to specific groups that the match AS paths defined in an AS-path access list. All other local sources will not be advertised in SA messages.
- You can configure the device to only originate SA messages for local sources that match the criteria defined in the route map. All other local sources will not be advertised in SA messages.
- You configure an SA origination filter that includes an extended access list, an AS-path access list, and route map, or a combination thereof. In this case, all conditions must be true before any local sources are advertised in SA messages.

Use of Outgoing Filter Lists in MSDP

By default, an MSDP-enabled device forwards all SA messages it receives to all of its MSDP peers. However, you can prevent SA messages from being forwarded to MSDP peers by creating outgoing filter lists. Outgoing filter lists apply to all SA messages, whether locally originated or received from another MSDP peer, whereas SA origination filters apply only to locally originated SA messages. For more information about enabling a filter for MSDP SA messages originated by the local device, see the [Controlling SA Messages Originated by an RP for Local Sources](#) section.

By creating an outgoing filter list, you can control the SA messages that a device forwards to a peer as follows:

- You can filter all outgoing SA messages forwarded to a specified MSDP peer by configuring the device to stop forwarding its SA messages to the MSDP peer.
- You can filter a subset of outgoing SA messages forwarded to a specified MSDP peer based on (S, G) pairs defined in an extended access list by configuring the device to only forward SA messages to the MSDP peer that match the (S, G) pairs permitted in an extended access list. The forwarding of all other SA messages to the MSDP peer will be stopped.
- You can filter a subset of outgoing SA messages forwarded to a specified MSDP peer based on match criteria defined in a route map by configuring the device to only forward SA messages that match the criteria defined in the route map. The forwarding of all other SA messages to the MSDP peer will be stopped.
- You can filter a subset of outgoing SA messages from a specified peer based on the announcing RP address contained in the SA message by configuring the device to filter outgoing SA messages based on their origin, even after an SA message has been transmitted across one or more MSDP peers. The forwarding of all other SA messages to the MSDP peer will be stopped.
- You can configure an outgoing filter list that includes an extended access list, a route map, and either an RP access list or an RP route map. In this case, all conditions must be true for the MSDP peer to forward the outgoing SA message.

**Caution**

Arbitrary filtering of SA messages can result in downstream MSDP peers being starved of SA messages for legitimate active sources. Care, therefore, should be taken when using these sorts of filters. Normally, outgoing filter lists are used only to reject undesirable sources, such as sources using private addresses.

Use of Incoming Filter Lists in MSDP

By default, an MSDP-enabled device receives all SA messages sent to it from its MSDP peers. However, you can control the source information that a device receives from its MSDP peers by creating incoming filter lists.

By creating incoming filter lists, you can control the incoming SA messages that a device receives from its peers as follows:

- You can filter all incoming SA messages from a specified MSDP peer by configuring the device to ignore all SA messages sent to it from the specified MSDP peer.
- You can filter a subset of incoming SA messages from a specified peer based on (S, G) pairs defined in an extended access list by configuring the device to only receive SA messages from the MSDP peer that match the (S, G) pairs defined in the extended access list. All other incoming SA messages from the MSDP peer will be ignored.
- You can filter a subset of incoming SA request messages from a specified peer based on match criteria defined in a route map by configuring the device to only receive SA messages that match the criteria defined in the route map. All other incoming SA messages from the MSDP peer will be ignored.
- You can filter a subset of incoming SA messages from a specified peer based on both (S, G) pairs defined in an extended access list and on match criteria defined in a route map by configuring the device to only receive incoming SA messages that both match the (S, G) pairs defined in the extended access list and

match the criteria defined in the route map. All other incoming SA messages from the MSDP peer will be ignored.

- You can filter a subset of incoming SA messages from a specified peer based on the announcing RP address contained in the SA message by configuring the device to filter incoming SA messages based on their origin, even after the SA message may have already been transmitted across one or more MSDP peers.
- You can configure an incoming filter list that includes an extended access list, a route map, and either an RP access list or an RP route map. In this case, all conditions must be true for the MSDP peer to receive the incoming SA message.



Caution Arbitrary filtering of SA messages can result in downstream MSDP peers being starved of SA messages for legitimate active sources. Care, therefore, should be taken when using these sorts of filters. Normally, incoming filter lists are used only to reject undesirable sources, such as sources using private addresses.

TTL Thresholds in MSDP

The time-to-live (TTL) value provides a means to limit the number of hops a packet can take before being dropped. The **ip multicast ttl-threshold** command is used to specify a TTL for data-encapsulated SA messages sent to specified MSDP peers. By default, multicast data packets in SA messages are sent to an MSDP peer, provided the TTL value of the packet is greater than 0, which is standard TTL behavior.

In general, a TTL-threshold problem can be introduced by the encapsulation of a source's initial multicast packet in an SA message. Because the multicast packet is encapsulated inside of the unicast SA message (whose TTL is 255), its TTL is not decremented as the SA message travels to the MSDP peer. Furthermore, the total number of hops that the SA message traverses can be drastically different than a normal multicast packet because multicast and unicast traffic may follow completely different paths to the MSDP peer and hence the remote PIM-SM domain. As a result, encapsulated packets can end up violating TTL thresholds. The solution to this problem is to configure a TTL threshold that is associated with any multicast packet that is encapsulated in an SA message sent to a particular MSDP peer using the **ip multicast ttl-threshold** command. The **ip msdp ttl-threshold** command prevents any multicast packet whose TTL in the IP header is less than the TTL value specified for the *ttl-value* argument from being encapsulated in SA messages sent to that peer.

MSDP Message Types

There are four basic MSDP message types, each encoded in their own Type, Length, and Value (TLV) data format.

SA Messages

SA messages are used to advertise active sources in a domain. In addition, these SA messages may contain the initial multicast data packet that was sent by the source.

SA messages contain the IP address of the originating RP and one or more (S, G) pairs being advertised. In addition, the SA message may contain an encapsulated data packet.

SA Request Messages

SA request messages are used to request a list of active sources for a specific group. These messages are sent to an MSDP SA cache that maintains a list of active (S, G) pairs in its SA cache. Join latency can be reduced by using SA request messages to request the list of active sources for a group instead of having to wait up to 60 seconds for all active sources in the group to be readvertised by originating RPs.

SA Response Messages

SA response messages are sent by the MSDP peer in response to an SA request message. SA response messages contain the IP address of the originating RP and one or more (S, G) pairs of the active sources in the originating RP's domain that are stored in the cache.

Keepalive Messages

Keepalive messages are sent every 60 seconds in order to keep the MSDP session active. If no keepalive messages or SA messages are received for 75 seconds, the MSDP session is reset.

Default MSDP Configuration

MSDP is not enabled, and no default MSDP peer exists.

How to Configure MSDP

Configuring a Default MSDP Peer

Before you begin

Configure an MSDP peer.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip msdp default-peer <i>ip-address</i> <i>name</i> [prefix-list <i>list</i>] Example:	Defines a default peer from which to accept all MSDP SA messages.

	Command or Action	Purpose
	<pre>Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a</pre>	<ul style="list-style-type: none"> For <i>ip-address / name</i>, enter the IP address or Domain Name System (DNS) server name of the MSDP default peer. (Optional) For prefix-list list, enter the list name that specifies the peer to be the default peer only for the listed prefixes. You can have multiple active default peers when you have a prefix list associated with each. <p>When you enter multiple ip msdp default-peer commands with the prefix-list keyword, you use all the default peers at the same time for different RP prefixes. This syntax is typically used in a service provider cloud that connects stub site clouds.</p> <p>When you enter multiple ip msdp default-peer commands without the prefix-list keyword, a single active peer accepts all SA messages. If that peer fails, the next configured default peer accepts all SA messages. This syntax is typically used at a stub site.</p>
Step 4	<p>ip prefix-list <i>name</i> [description <i>string</i>] seq <i>number</i> {permit deny} <i>network length</i></p> <p>Example:</p> <pre>Router(config)# prefix-list site-a seq 3 permit 12 network length 128</pre>	<p>(Optional) Creates a prefix list using the name specified in Step 2.</p> <ul style="list-style-type: none"> (Optional) For description string, enter a description of up to 80 characters to describe this prefix list. For seq number, enter the sequence number of the entry. The range is 1 to 4294967294. The deny keyword denies access to matching conditions. The permit keyword permits access to matching conditions. For <i>network length</i>, specify the network number and length (in bits) of the network mask that is permitted or denied.
Step 5	<p>ip msdp description {<i>peer-name</i> <i>peer-address</i>} <i>text</i></p> <p>Example:</p> <pre>Router(config)# ip msdp description peer-name site-b</pre>	<p>(Optional) Configures a description for the specified peer to make it easier to identify in a configuration or in show command output.</p> <p>By default, no description is associated with an MSDP peer.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
Step 7	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Caching Source-Active State

If you want to sacrifice some memory in exchange for reducing the latency of the source information, you can configure the Device to cache SA messages. Perform the following steps to enable the caching of source/group pairs:

Follow these steps to enable the caching of source/group pairs:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip msdp cache-sa-state [list access-list-number] Example: <pre>Switch(config)# ip msdp cache-sa-state 100</pre>	Enables the caching of source/group pairs (create an SA state). Those pairs that pass the access list are cached. For list access-list-number , the range is 100 to 199. Note An alternative to this command is the ip msdp sa-reques global configuration command, which causes the Device to send an SA request message to the MSDP peer when a new member for a group becomes active.
Step 4	access-list access-list-number {deny permit} protocol source source-wildcard destination destination-wildcard	Creates an IP extended access list, repeating the command as many times as necessary.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255</pre>	<ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 100 to 199. Enter the same number created in Step 2. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>protocol</i>, enter ip as the protocol name. • For <i>source</i>, enter the number of the network or host from which the packet is being sent. • For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For <i>destination</i>, enter the number of the network or host to which the packet is being sent. • For <i>destination-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Requesting Source Information from an MSDP Peer

If you want a new member of a group to learn the active multicast sources in a connected PIM sparse-mode domain that are sending to a group, perform this task for the Device to send SA request messages to the specified MSDP peer when a new member joins a group. The peer replies with the information in its SA cache. If the peer does not have a cache configured, this command has no result. Configuring this feature reduces join latency but sacrifices memory.

Follow these steps to configure the Device to send SA request messages to the MSDP peer when a new member joins a group and wants to receive multicast traffic:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip msdp sa-request <i>{ip-address name}</i> Example: Switch(config)# ip msdp sa-request 171.69.1.1	Configure the Device to send SA request messages to the specified MSDP peer. For <i>ip-address name</i> , enter the IP address or name of the MSDP peer from which the local Device requests SA messages when a new member for a group becomes active. Repeat the command for each MSDP peer that you want to supply with SA messages.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Controlling Source Information that Your Switch Originates

You can control the multicast source information that originates with your Device:

- Sources you advertise (based on your sources)

- Receivers of source information (based on knowing the requestor)

For more information, see the [Redistributing Sources, on page 265](#) and the [Filtering Source-Active Request Messages, on page 267](#).

Redistributing Sources

SA messages originate on RPs to which sources have registered. By default, any source that registers with an RP is advertised. The *A flag* is set in the RP when a source is registered, which means the source is advertised in an SA unless it is filtered.

Follow these steps to further restrict which registered sources are advertised:

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip msdp redistribute [list <i>access-list-name</i>] [asn <i>aspath-access-list-number</i>] [route-map <i>map</i>]</p> <p>Example:</p> <pre>Switch(config)# ip msdp redistribute list 21</pre>	<p>Configures which (S,G) entries from the multicast routing table are advertised in SA messages.</p> <p>By default, only sources within the local domain are advertised.</p> <ul style="list-style-type: none"> • (Optional) list <i>access-list-name</i>— Enters the name or number of an IP standard or extended access list. The range is 1 to 99 for standard access lists and 100 to 199 for extended lists. The access list controls which local sources are advertised and to which groups they send. • (Optional) asn <i>aspath-access-list-number</i>—Enters the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the ip as-path access-list command. • (Optional) route-map <i>map</i>—Enters the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the ip as-path access-list command. <p>The Device advertises (S,G) pairs according to the access list or autonomous system path access list.</p>

	Command or Action	Purpose
Step 4	<p>Use one of the following:</p> <ul style="list-style-type: none"> • <code>access-list</code><i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>] • <code>access-list</code><i>access-list-number</i> {deny permit} <i>protocol source source-wildcard destination</i> <i>destination-wildcard</i> <p>Example:</p> <pre>Switch(config)# access list 21 permit 194.1.22.0</pre> <p>or</p> <pre>Switch(config)# access list 21 permit ip 194.1.22.0 10.1.1.10 194.3.44.0 10.1.1.10</pre>	<p>Creates an IP standard access list, repeating the command as many times as necessary.</p> <p>or</p> <p>Creates an IP extended access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • <i>access-list-number</i>—Enters the same number created in Step 2. The range is 1 to 99 for standard access lists and 100 to 199 for extended lists. • deny—Denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • <i>protocol</i>—Enters ip as the protocol name. • <i>source</i>—Enters the number of the network or host from which the packet is being sent. • <i>source-wildcard</i>—Enters the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • <i>destination</i>—Enters the number of the network or host to which the packet is being sent. • <i>destination-wildcard</i>—Enters the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Filtering Source-Active Request Messages

By default, only Device that are caching SA information can respond to SA requests. By default, such a Device honors all SA request messages from its MSDP peers and supplies the IP addresses of the active sources.

However, you can configure the Device to ignore all SA requests from an MSDP peer. You can also honor only those SA request messages from a peer for groups described by a standard access list. If the groups in the access list pass, SA request messages are accepted. All other such messages from the peer for other groups are ignored.

To return to the default setting, use the **no ip msdp filter-sa-request** *{ip-address| name}* global configuration command.

Follow these steps to configure one of these options:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • <code>ip msdp filter-sa-request {ip-addressname}</code> • <code>ip msdp filter-sa-request {ip-addressname} list access-list-number</code> Example: <pre>Switch(config)# ip msdp filter sa-request 171.69.2.2</pre>	Filters all SA request messages from the specified MSDP peer. or Filters SA request messages from the specified MSDP peer for groups that pass the standard access list. The access list describes a multicast group address. The range for the access-list-number is 1 to 99.
Step 4	access-list access-list-number {deny permit} source [source-wildcard] Example: <pre>Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255</pre>	Creates an IP standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>source</i>, enter the number of the network or host from which the packet is being sent.

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 5	end Example: <pre>Switch(config) # end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Controlling Source Information that Your Switch Forwards

By default, the Device forwards all SA messages it receives to all its MSDP peers. However, you can prevent outgoing messages from being forwarded to a peer by using a filter or by setting a time-to-live (TTL) value.

Using a Filter

By creating a filter, you can perform one of these actions:

- Filter all source/group pairs
- Specify an IP extended access list to pass only certain source/group pairs
- Filter based on match criteria in a route map

Follow these steps to apply a filter:

Procedure

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
	Switch> <code>enable</code>	
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>Use one of the following:</p> <ul style="list-style-type: none"> • ip msdp sa-filter out <pre>{ip-address name}</pre> <ul style="list-style-type: none"> • ip msdp sa-filter out <pre>{ip-address name} list access-list-number</pre> <ul style="list-style-type: none"> • ip msdp sa-filter out <pre>{ip-address name} route-map map-tag</pre> <p>Example:</p> <pre>Switch(config)# ip msdp sa-filter out switch.cisco.com</pre> <p>OR</p> <pre>Switch(config)# ip msdp sa-filter out list 100</pre> <p>OR</p> <pre>Switch(config)# ip msdp sa-filter out switch.cisco.com route-map 22</pre>	<ul style="list-style-type: none"> • Filters all SA messages to the specified MSDP peer. • Passes only those SA messages that pass the IP extended access list to the specified peer. The range for the extended <i>access-list-number</i> is 100 to 199. <p>If both the list and the route-map keywords are used, all conditions must be true to pass any (S,G) pair in outgoing SA messages.</p> <ul style="list-style-type: none"> • Passes only those SA messages that meet the match criteria in the route map <i>map-tag</i> to the specified MSDP peer. <p>If all match criteria are true, a permit from the route map passes routes through the filter. A deny filters routes.</p>
Step 4	<p>access-list access-list-number {deny permit} protocol source source-wildcard destination destination-wildcard</p> <p>Example:</p> <pre>Switch(config)# access list 100 permit ip 194.1.22.0 10.1.1.10 194.3.44.0 10.1.1.10</pre>	<p>(Optional) Creates an IP extended access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the number specified in Step 2. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>protocol</i>, enter ip as the protocol name. • For <i>source</i>, enter the number of the network or host from which the packet is being sent.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. For <i>destination</i>, enter the number of the network or host to which the packet is being sent. For <i>destination-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Using TTL to Limit the Multicast Data Sent in SA Messages

You can use a TTL value to control what data is encapsulated in the first SA message for every source. Only multicast packets with an IP-header TTL greater than or equal to the *ttl* argument are sent to the specified MSDP peer. For example, you can limit internal traffic to a TTL of 8. If you want other groups to go to external locations, you must send those packets with a TTL greater than 8.

Follow these steps to establish a TTL threshold:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip msdp ttl-threshold { <i>ip-address</i> <i>name</i> } <i>ttl</i> Example: <pre>Switch(config)# ip msdp ttl-threshold switch.cisco.com 0</pre>	Limits which multicast data is encapsulated in the first SA message to the specified MSDP peer. <ul style="list-style-type: none"> • For <i>ip-address</i> <i>name</i>, enter the IP address or name of the MSDP peer to which the TTL limitation applies. • For <i>ttl</i>, enter the TTL value. The default is 0, which means all multicast data packets are forwarded to the peer until the TTL is exhausted. The range is 0 to 255.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Controlling Source Information that Your Switch Receives

By default, the Device receives all SA messages that its MSDP RPF peers send to it. However, you can control the source information that you receive from MSDP peers by filtering incoming SA messages. In other words, you can configure the Device to not accept them.

You can perform one of these actions:

- Filter all incoming SA messages from an MSDP peer
- Specify an IP extended access list to pass certain source/group pairs
- Filter based on match criteria in a route map

Follow these steps to apply a filter:

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>Use one of the following:</p> <ul style="list-style-type: none"> • ip msdp sa-filter in <pre>{ip-address name}</pre> <ul style="list-style-type: none"> • ip msdp sa-filter in <pre>{ip-address name} list access-list-number</pre> <ul style="list-style-type: none"> • ip msdp sa-filter in <pre>{ip-address name} route-map map-tag</pre> <p>Example:</p> <pre>Switch(config)# ip msdp sa-filter in switch.cisco.com</pre> <p>or</p> <pre>Switch(config)# ip msdp sa-filter in list 100</pre> <p>or</p> <pre>Switch(config)# ip msdp sa-filter in switch.cisco.com route-map 22</pre>	<ul style="list-style-type: none"> • Filters all SA messages to the specified MSDP peer. • Passes only those SA messages from the specified peer that pass the IP extended access list. The range for the extended <i>access-list-number</i> is 100 to 199. <p>If both the list and the route-map keywords are used, all conditions must be true to pass any (S,G) pair in outgoing SA messages.</p> <ul style="list-style-type: none"> • Passes only those SA messages from the specified MSDP peer that meet the match criteria in the route map <i>map-tag</i>. <p>If all match criteria are true, a permit from the route map passes routes through the filter. A deny filters routes.</p>
Step 4	<p>access-list <i>access-list-number</i> {deny permit} <i>protocol source source-wildcard destination destination-wildcard</i></p> <p>Example:</p> <pre>Switch(config)# access list 100 permit ip 194.1.22.0 10.1.1.10 194.3.44.0 10.1.1.10</pre>	<p>(Optional) Creates an IP extended access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • <i>access-list-number</i>, enter the number specified in Step 2. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>protocol</i>, enter ip as the protocol name.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>source</i>, enter the number of the network or host from which the packet is being sent. For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. For <i>destination</i>, enter the number of the network or host to which the packet is being sent. For <i>destination-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring an MSDP Mesh Group

Perform this optional task to configure an MSDP mesh group.



Note You can configure multiple mesh groups per device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip msdp mesh-group** *mesh-name* {*peer-address* | *peer-name*}
4. Repeat Step 3 to add MSDP peers as members of the mesh group.

5. `exit`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip msdp mesh-group <i>mesh-name</i> { <i>peer-address</i> <i>peer-name</i> } Example: <pre>Device(config)# ip msdp mesh-group peermesh</pre>	Configures an MSDP mesh group and indicates that an MSDP peer belongs to that mesh group. Note All MSDP peers on a device that participate in a mesh group must be fully meshed with all other MSDP peers in the group. Each MSDP peer on each device must be configured as a peer using the ip msdp peer command and also as a member of the mesh group using the ip msdp mesh-group command.
Step 4	Repeat Step 3 to add MSDP peers as members of the mesh group.	--
Step 5	exit Example: <pre>Device(config)# exit</pre>	Exits global configuration mode and returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Shutting Down an MSDP Peer

Before you begin

MSDP is running and the MSDP peers must be configured.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip msdp shutdown** *{peer-name | peer-address}*
4. Repeat Step 3 to shut down additional MSDP peers.
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip msdp shutdown <i>{peer-name peer-address}</i> Example: Device(config)# ip msdp shutdown 192.168.1.3	Administratively shuts down the specified MSDP peer.
Step 4	Repeat Step 3 to shut down additional MSDP peers.	--
Step 5	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Including a Bordering PIM Dense-Mode Region in MSDP

You can configure MSDP on a Device that borders a PIM sparse-mode region with a dense-mode region. By default, active sources in the dense-mode region do not participate in MSDP.



Note We do not recommend using the `ip msdp border sa-address` global configuration command. It is better to configure the border router in the sparse-mode domain to proxy-register sources in the dense-mode domain to the RP of the sparse-mode domain and have the sparse-mode domain use standard MSDP procedures to advertise these sources.

The `ip msdp originator-id` global configuration command also identifies an interface to be used as the RP address. If both the `ip msdp border sa-address` and the `ip msdp originator-id` global configuration commands are configured, the address derived from the `ip msdp originator-id` command specifies the RP address.

Follow these steps to configure the border router to send SA messages for sources active in the dense-mode region to the MSDP peers:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip msdp border sa-address <i>interface-id</i> Example: Switch(config)# <code>ip msdp border sa-address 0/1</code>	Configures the switch on the border between a dense-mode and sparse-mode region to send SA messages about active sources in the dense-mode region. For <i>interface-id</i> , specifies the interface from which the IP address is derived and used as the RP address in SA messages.

	Command or Action	Purpose
		The IP address of the interface is used as the Originator-ID, which is the RP field in the SA message.
Step 4	ip msdp redistribute [<i>list access-list-name</i>] [<i>asn aspath-access-list-number</i>] [route-map map] Example: Switch(config)# ip msdp redistribute list 100	Configures which (S,G) entries from the multicast routing table are advertised in SA messages. For more information, see the Redistributing Sources, on page 265 .
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring an Originating Address other than the RP Address

Perform this optional task to allow an MSDP speaker that originates an SA message to use the IP address of its interface as the RP address in the SA message.

You can also change the originator ID for any one of the following reasons:

- If you configure multiple devices in an MSDP mesh group for Anycast RP.
- If you have a device that borders a PIM-SM domain and a PIM-DM domain. If a device borders a PIM-SM domain and a PIM-DM domain and you want to advertise active sources within the PIM-DM domain, configure the RP address in SA messages to be the address of the originating device's interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip msdp originator-id**
4. **exit**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip msdp originator-id Example: Switch(config)# ip msdp originator-id ethernet 1	Configures the RP address in SA messages to be the address of the originating device's interface.
Step 4	exit Example: Switch(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining MSDP

Monitoring MSDP

Perform this optional task to monitor MSDP SA messages, peers, state, and peer status.

SUMMARY STEPS

1. enable
2. debug ip msdp [*peer-address* | *peer-name*] [**detail**] [**routes**]
3. debug ip msdp resets

4. **show ip msdp count** [*as-number*]
5. **show ip msdp peer** [*peer-address* | *peer-name*]
6. **show ip msdp sa-cache** [*group-address* | *source-address* | *group-name* | *source-name*] [*as-number*]
7. **show ip msdp summary**

DETAILED STEPS

Step 1 enable

Example:

```
Device# enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 debug ip msdp [*peer-address* | *peer-name*] [**detail**] [**routes**]

Use this command to debug MSDP activity.

Use the optional *peer-address* or *peer-name* argument to specify for which peer debug events are logged.

The following is sample output from the **debug ip msdp** command:

Example:

```
Device# debug ip msdp
MSDP debugging is on
Device#
MSDP: 224.150.44.254: Received 1388-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 1388, ec: 115, RP: 172.31.3.92
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.92, used EMBGP peer
MSDP: 224.150.44.250: Forward 1388-byte SA to peer
MSDP: 224.150.44.254: Received 1028-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 1028, ec: 85, RP: 172.31.3.92
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.92, used EMBGP peer
MSDP: 224.150.44.250: Forward 1028-byte SA to peer
MSDP: 224.150.44.254: Received 1388-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 1388, ec: 115, RP: 172.31.3.111
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.111, used EMBGP peer
MSDP: 224.150.44.250: Forward 1388-byte SA to peer
MSDP: 224.150.44.250: Received 56-byte message from peer
MSDP: 224.150.44.250: SA TLV, len: 56, ec: 4, RP: 192.168.76.241
MSDP: 224.150.44.250: Peer RPF check passed for 192.168.76.241, used EMBGP peer
MSDP: 224.150.44.254: Forward 56-byte SA to peer
MSDP: 224.150.44.254: Received 116-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 116, ec: 9, RP: 172.31.3.111
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.111, used EMBGP peer
MSDP: 224.150.44.250: Forward 116-byte SA to peer
MSDP: 224.150.44.254: Received 32-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 32, ec: 2, RP: 172.31.3.78
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.78, used EMBGP peer
MSDP: 224.150.44.250: Forward 32-byte SA to peer
```

Step 3 debug ip msdp resets

Use this command to debug MSDP peer reset reasons.

Example:

```
Device# debug ip msdp resets
```

Step 4 **show ip msdp count** [*as-number*]

Use this command to display the number of sources and groups originated in MSDP SA messages and the number of SA messages from an MSDP peer in the SA cache. The **ip msdp cache-sa-state** command must be configured for this command to produce any output.

The following is sample output from the **show ip msdp count** command:

Example:

```
Device# show ip msdp count
SA State per Peer Counters, <Peer>: <# SA learned>
 192.168.4.4: 8
SA State per ASN Counters, <asn>: <# sources>/<# groups>
 Total entries: 8
?: 8/8
```

Step 5 **show ip msdp peer** [*peer-address* | *peer-name*]

Use this command to display detailed information about MSDP peers.

Use the optional *peer-address* or *peer-name* argument to display information about a particular peer.

The following is sample output from the **show ip msdp peer** command:

Example:

```
Device# show ip msdp peer 192.168.4.4
MSDP Peer 192.168.4.4 (?), AS 64512 (configured AS)
 Connection status:
  State: Up, Resets: 0, Connection source: Loopback0 (2.2.2.2)
  Uptime(Downtime): 00:07:55, Messages sent/received: 8/18
  Output messages discarded: 0
  Connection and counters cleared 00:08:55 ago
SA Filtering:
  Input (S,G) filter: none, route-map: none
  Input RP filter: none, route-map: none
  Output (S,G) filter: none, route-map: none
  Output RP filter: none, route-map: none
SA-Requests:
  Input filter: none
Peer ttl threshold: 0
SAs learned from this peer: 8
Input queue size: 0, Output queue size: 0
MD5 signature protection on MSDP TCP connection: not enabled
```

Step 6 **show ip msdp sa-cache** [*group-address* | *source-address* | *group-name* | *source-name*] [*as-number*]

Use this command to display the (S, G) state learned from MSDP peers.

The following is sample output from the **show ip msdp sa-cache** command:

Example:

```
Device# show ip msdp sa-cache
MSDP Source-Active Cache - 8 entries
(10.44.44.5, 239.232.1.0), RP 192.168.4.4, BGP/AS 64512, 00:01:20/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.1), RP 192.168.4.4, BGP/AS 64512, 00:01:20/00:05:32, Peer 192.168.4.4
```



```
(10.44.44.5, 239.232.1.2), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.3), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.4), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.5), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.6), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.7), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
```

Step 7 show ip msdp summary

Use this command to display MSDP peer status.

The following is sample output from the **show ip msdp summary** command:

Example:

```
Device# show ip msdp summary
MSDP Peer Status Summary
Peer Address      AS      State   Uptime/  Reset SA   Peer Name
                  AS      State   Downtime Count Count
192.168.4.4       4       Up      00:08:05 0       8       ?
```

Clearing MSDP Connections Statistics and SA Cache Entries

Perform this optional task to clear MSDP connections, statistics, and SA cache entries.

SUMMARY STEPS

1. **enable**
2. **clear ip msdp peer** [*peer-address* | *peer-name*]
3. **clear ip msdp statistics** [*peer-address* | *peer-name*]
4. **clear ip msdp sa-cache** [*group-address*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	clear ip msdp peer [<i>peer-address</i> <i>peer-name</i>] Example: Device# clear ip msdp peer	Clears the TCP connection to the specified MSDP peer and resets all MSDP message counters.
Step 3	clear ip msdp statistics [<i>peer-address</i> <i>peer-name</i>] Example: Device# clear ip msdp statistics	Clears the statistics counters for the specified MSDP peer and resets all MSDP message counters.
Step 4	clear ip msdp sa-cache [<i>group-address</i>]	Clears SA cache entries.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device# clear ip msdp sa-cache</pre>	<ul style="list-style-type: none"> • If the clear ip msdp sa-cache is specified with the optional <i>group-address</i> argument or <i>source-address</i> argument, all SA cache entries are cleared. • Use the optional <i>group-address</i> argument to clear all SA cache entries associated with a specific group.

Configuration Examples for Configuring MSDP

Configuring a Default MSDP Peer: Example

This example shows a partial configuration of Router A and Router C in . Each of these ISPs have more than one customer (like the customer in) who use default peering (no BGP or MBGP). In that case, they might have similar configurations. That is, they accept SAs only from a default peer if the SA is permitted by the corresponding prefix list.

Router A

```
Router(config)# ip msdp default-peer 10.1.1.1
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Router C

```
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Caching Source-Active State: Example

This example shows how to enable the cache state for all sources in 171.69.0.0/16 sending to groups 224.2.0.0/16:

```
Switch(config)# ip msdp cache-sa-state 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255
```

Requesting Source Information from an MSDP Peer: Example

This example shows how to configure the switch to send SA request messages to the MSDP peer at 171.69.1.1:

```
Switch(config)# ip msdp sa-request 171.69.1.1
```

Controlling Source Information that Your Switch Originates: Example

This example shows how to configure the switch to filter SA request messages from the MSDP peer at 171.69.2.2. SA request messages from sources on network 192.4.22.0 pass access list 1 and are accepted; all others are ignored.

```
Switch(config)# ip msdp filter sa-request 171.69.2.2 list 1
Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255
```

Controlling Source Information that Your Switch Forwards: Example

This example shows how to allow only (S,G) pairs that pass access list 100 to be forwarded in an SA message to the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet1/0/1
Switch(config)# ip msdp sa-filter out switch.cisco.com list 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.20 0 0.0.255.255
```

Controlling Source Information that Your Switch Receives: Example

This example shows how to filter all SA messages from the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet1/0/1
Switch(config)# ip msdp sa-filter in switch.cisco.com
```

Example: Configuring MSDP Mesh Groups

The following example shows how to configure three devices to be fully meshed members of an MSDP mesh group:

Device A Configuration

```
ip msdp peer 10.2.2.2
ip msdp peer 10.3.3.3
ip msdp mesh-group test-mesh-group 10.2.2.2
ip msdp mesh-group test-mesh-group 10.3.3.3
```

Device B Configuration

```
ip msdp peer 10.1.1.1
ip msdp peer 10.3.3.3
ip msdp mesh-group test-mesh-group 10.1.1.1
ip msdp mesh-group test-mesh-group 10.3.3.3
```

Device C Configuration

```
ip msdp peer 10.1.1.1
ip msdp peer 10.2.2.2
```

```
ip msdp mesh-group test-mesh-group 10.1.1.1
ip msdp mesh-group test-mesh-group 10.2.2.2
```

Requesting Source Information from an MSDP Peer: Example

This example shows how to configure the switch to send SA request messages to the MSDP peer at 171.69.1.1:

```
Switch(config)# ip msdp sa-request 171.69.1.1
```



PART **III**

IPv6

- [Configuring MLD Snooping, on page 287](#)
- [Configuring IPv6 Unicast Routing, on page 303](#)
- [Implementing IPv6 Multicast, on page 357](#)



CHAPTER 20

Configuring MLD Snooping

This module contains details of configuring MLD snooping

- [Finding Feature Information, on page 287](#)
- [Information About Configuring IPv6 MLD Snooping, on page 287](#)
- [How to Configure IPv6 MLD Snooping, on page 291](#)
- [Displaying MLD Snooping Information, on page 299](#)
- [Configuration Examples for Configuring MLD Snooping, on page 300](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Information About Configuring IPv6 MLD Snooping

You can use Multicast Listener Discovery (MLD) snooping to enable efficient distribution of IP Version 6 (IPv6) multicast data to clients and routers in a switched network on the switch. Unless otherwise noted, the term switch refers to a standalone switch.



Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release or the Cisco IOS documentation referenced in the procedures.

Understanding MLD Snooping

In IP Version 4 (IPv4), Layer 2 switches can use Internet Group Management Protocol (IGMP) snooping to limit the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. In IPv6, MLD snooping performs

a similar function. With MLD snooping, IPv6 multicast data is selectively forwarded to a list of ports that want to receive the data, instead of being flooded to all ports in a VLAN. This list is constructed by snooping IPv6 multicast control packets.

MLD is a protocol used by IPv6 multicast routers to discover the presence of multicast listeners (nodes wishing to receive IPv6 multicast packets) on the links that are directly attached to the routers and to discover which multicast packets are of interest to neighboring nodes. MLD is derived from IGMP; MLD Version 1 (MLDv1) is equivalent to IGMPv2, and MLD Version 2 (MLDv2) is equivalent to IGMPv3. MLD is a subprotocol of Internet Control Message Protocol Version 6 (ICMPv6), and MLD messages are a subset of ICMPv6 messages, identified in IPv6 packets by a preceding Next Header value of 58.

The switch supports two versions of MLD snooping:

- MLDv1 snooping detects MLDv1 control packets and sets up traffic bridging based on IPv6 destination multicast addresses.
- MLDv2 basic snooping (MBSS) uses MLDv2 control packets to set up traffic forwarding based on IPv6 destination multicast addresses.

The switch can snoop on both MLDv1 and MLDv2 protocol packets and bridge IPv6 multicast data based on destination IPv6 multicast addresses.



Note The switch does not support MLDv2 enhanced snooping, which sets up IPv6 source and destination multicast address-based forwarding.

MLD snooping can be enabled or disabled globally or per VLAN. When MLD snooping is enabled, a per-VLAN IPv6 multicast address table is constructed in software and hardware. The switch then performs IPv6 multicast-address based bridging in hardware.

MLD Messages

MLDv1 supports three types of messages:

- Listener Queries are the equivalent of IGMPv2 queries and are either General Queries or Multicast-Address-Specific Queries (MASQs).
- Multicast Listener Reports are the equivalent of IGMPv2 reports
- Multicast Listener Done messages are the equivalent of IGMPv2 leave messages.

MLDv2 supports MLDv2 queries and reports, as well as MLDv1 Report and Done messages.

Message timers and state transitions resulting from messages being sent or received are the same as those of IGMPv2 messages. MLD messages that do not have valid link-local IPv6 source addresses are ignored by MLD routers and switches.

MLD Queries

The switch sends out MLD queries, constructs an IPv6 multicast address database, and generates MLD group-specific and MLD group-and-source-specific queries in response to MLD Done messages. The switch also supports report suppression, report proxying, Immediate-Leave functionality, and static IPv6 multicast group address configuration.

When MLD snooping is disabled, all MLD queries are flooded in the ingress VLAN.

When MLD snooping is enabled, received MLD queries are flooded in the ingress VLAN, and a copy of the query is sent to the CPU for processing. From the received query, MLD snooping builds the IPv6 multicast address database. It detects multicast router ports, maintains timers, sets report response time, learns the querier IP source address for the VLAN, learns the querier port in the VLAN, and maintains multicast-address aging.



Note When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for the Catalyst 2960, 2960-S, 2960-C, 2960-X or 2960-CX switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.

When a group exists in the MLD snooping database, the switch responds to a group-specific query by sending an MLDv1 report. When the group is unknown, the group-specific query is flooded to the ingress VLAN.

When a host wants to leave a multicast group, it can send out an MLD Done message (equivalent to IGMP Leave message). When the switch receives an MLDv1 Done message, if Immediate-Leave is not enabled, the switch sends an MASQ to the port from which the message was received to determine if other devices connected to the port should remain in the multicast group.

Multicast Client Aging Robustness

You can configure port membership removal from addresses based on the number of queries. A port is removed from membership to an address only when there are no reports to the address on the port for the configured number of queries. The default number is 2.

Multicast Router Discovery

Like IGMP snooping, MLD snooping performs multicast router discovery, with these characteristics:

- Ports configured by a user never age out.
- Dynamic port learning results from MLDv1 snooping queries and IPv6 PIMv2 packets.
- If there are multiple routers on the same Layer 2 interface, MLD snooping tracks a single multicast router on the port (the router that most recently sent a router control packet).
- Dynamic multicast router port aging is based on a default timer of 5 minutes; the multicast router is deleted from the router port list if no control packet is received on the port for 5 minutes.
- IPv6 multicast router discovery only takes place when MLD snooping is enabled on the switch.
- Received IPv6 multicast router control packets are always flooded to the ingress VLAN, whether or not MLD snooping is enabled on the switch.
- After the discovery of the first IPv6 multicast router port, unknown IPv6 multicast data is forwarded only to the discovered router ports (before that time, all IPv6 multicast data is flooded to the ingress VLAN).

MLD Reports

The processing of MLDv1 join messages is essentially the same as with IGMPv2. When no IPv6 multicast routers are detected in a VLAN, reports are not processed or forwarded from the switch. When IPv6 multicast routers are detected and an MLDv1 report is received, an IPv6 multicast group address is entered in the VLAN

MLD database. Then all IPv6 multicast traffic to the group within the VLAN is forwarded using this address. When MLD snooping is disabled, reports are flooded in the ingress VLAN.

When MLD snooping is enabled, MLD report suppression, called listener message suppression, is automatically enabled. With report suppression, the switch forwards the first MLDv1 report received by a group to IPv6 multicast routers; subsequent reports for the group are not sent to the routers. When MLD snooping is disabled, report suppression is disabled, and all MLDv1 reports are flooded to the ingress VLAN.

The switch also supports MLDv1 proxy reporting. When an MLDv1 MASQ is received, the switch responds with MLDv1 reports for the address on which the query arrived if the group exists in the switch on another port and if the port on which the query arrived is not the last member port for the address.

MLD Done Messages and Immediate-Leave

When the Immediate-Leave feature is enabled and a host sends an MLDv1 Done message (equivalent to an IGMP leave message), the port on which the Done message was received is immediately deleted from the group. You enable Immediate-Leave on VLANs and (as with IGMP snooping), you should only use the feature on VLANs where a single host is connected to the port. If the port was the last member of a group, the group is also deleted, and the leave information is forwarded to the detected IPv6 multicast routers.

When Immediate Leave is not enabled in a VLAN (which would be the case when there are multiple clients for a group on the same port) and a Done message is received on a port, an MASQ is generated on that port. The user can control when a port membership is removed for an existing address in terms of the number of MASQs. A port is removed from membership to an address when there are no MLDv1 reports to the address on the port for the configured number of queries.

The number of MASQs generated is configured by using the **ipv6 mld snooping last-listener-query count** global configuration command. The default number is 2.

The MASQ is sent to the IPv6 multicast address for which the Done message was sent. If there are no reports sent to the IPv6 multicast address specified in the MASQ during the switch maximum response time, the port on which the MASQ was sent is deleted from the IPv6 multicast address database. The maximum response time is the time configured by using the **ipv6 mld snooping last-listener-query-interval** global configuration command. If the deleted port is the last member of the multicast address, the multicast address is also deleted, and the switch sends the address leave information to all detected multicast routers.

Topology Change Notification Processing

When topology change notification (TCN) solicitation is enabled by using the **ipv6 mld snooping tcn query solicit** global configuration command, MLDv1 snooping sets the VLAN to flood all IPv6 multicast traffic with a configured number of MLDv1 queries before it begins sending multicast data only to selected ports. You set this value by using the **ipv6 mld snooping tcn flood query count** global configuration command. The default is to send two queries. The switch also generates MLDv1 global Done messages with valid link-local IPv6 source addresses when the switch becomes the STP root in the VLAN or when it is configured by the user. This is same as done in IGMP snooping.

MLD Snooping in Switch Stacks

The MLD IPv6 group address databases are maintained on all switches in the stack, regardless of which switch learns of an IPv6 multicast group. Report suppression and proxy reporting are done stack-wide. During the maximum response time, only one received report for a group is forwarded to the multicast routers, regardless of which switch the report arrives on.

The election of a new active stack does not affect the learning or bridging of IPv6 multicast data; bridging of IPv6 multicast data does not stop during an active stack re-election. When a new switch is added to the stack,

it synchronizes the learned IPv6 multicast information from the active stack . Until the synchronization is complete, data ingress on the newly added switch is treated as unknown multicast data.

How to Configure IPv6 MLD Snooping

Default MLD Snooping Configuration

Table 33: Default MLD Snooping Configuration

Feature	Default Setting
MLD snooping (Global)	Disabled.
MLD snooping (per VLAN)	Enabled. MLD snooping must be globally enabled for VLAN MLD snooping to take place.
IPv6 Multicast addresses	None configured.
IPv6 Multicast router ports	None configured.
MLD snooping Immediate Leave	Disabled.
MLD snooping robustness variable	Global: 2; Per VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query count	Global: 2; Per VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query interval	Global: 1000 (1 second); VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global interval.
TCN query solicit	Disabled.
TCN query count	2.
MLD listener suppression	Enabled.

MLD Snooping Configuration Guidelines

When configuring MLD snooping, consider these guidelines:

- You can configure MLD snooping characteristics at any time, but you must globally enable MLD snooping by using the **ipv6 mld snooping** global configuration command for the configuration to take effect.
- When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for the switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.
- MLD snooping and IGMP snooping act independently of each other. You can enable both features at the same time on the switch.

Enabling or Disabling MLD Snooping on the Switch

By default, IPv6 MLD snooping is globally disabled on the switch and enabled on all VLANs. When MLD snooping is globally disabled, it is also disabled on all VLANs. When you globally enable MLD snooping, the VLAN configuration overrides the global configuration. That is, MLD snooping is enabled only on VLAN interfaces in the default state (enabled).

You can enable and disable MLD snooping on a per-VLAN basis or for a range of VLANs, but if you globally disable MLD snooping, it is disabled in all VLANs. If global snooping is enabled, you can enable or disable VLAN snooping.

To globally enable MLD snooping on the switch, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 mld snooping Example: Switch(config)# ipv6 mld snooping	Enables MLD snooping on the switch.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	copy running-config startup-config Example: <pre>Switch(config)# copy running-config startup-config</pre>	(Optional) Save your entries in the configuration file.
Step 6	reload Example: <pre>Switch(config)# reload</pre>	Reload the operating system.

Enabling or Disabling MLD Snooping on a VLAN

To enable MLD snooping on a VLAN, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ipv6 mld snooping Example: <pre>Switch(config)# ipv6 mld snooping</pre>	Enables MLD snooping on the switch.
Step 4	ipv6 mld snooping vlan <i>vlan-id</i> Example: <pre>Switch(config)# ipv6 mld snooping vlan 1</pre>	Enables MLD snooping on the VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. Note MLD snooping must be globally enabled for VLAN snooping to be enabled.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# <code>ipv6 mld snooping vlan 1</code>	

Configuring a Static Multicast Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure an IPv6 multicast address and member ports for a VLAN.

To add a Layer 2 port as a member of a multicast group, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ipv6 mld snooping vlan <i>vlan-id</i> static <i>ipv6_multicast_address</i> interface <i>interface-id</i> Example: Switch(config)# <code>ipv6 mld snooping vlan 1 static 3333.0000.1111 interface gigabitethernet 0/1</code>	Configures a multicast group with a Layer 2 port as a member of a multicast group: <ul style="list-style-type: none"> • <i>vlan-id</i> is the multicast group VLAN ID. The VLAN ID range is 1 to 1001 and 1006 to 4094. • <i>ipv6_multicast_address</i> is the 128-bit group IPv6 address. The address must be in the form specified in RFC 2373. • <i>interface-id</i> is the member port. It can be a physical interface or a port channel (1 to 48).
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	Use one of the following: <ul style="list-style-type: none"> • <code>show ipv6 mld snooping address</code> • <code>show ipv6 mld snooping address vlan <i>vlan-id</i></code> Example:	Verifies the static member port and the IPv6 address.

	Command or Action	Purpose
	<pre>Switch# show ipv6 mld snooping address</pre> <p>OR</p> <pre>Switch# show ipv6 mld snooping vlan 1</pre>	

Configuring a Multicast Router Port



Note Static connections to multicast routers are supported only on switch ports.

To add a multicast router port to a VLAN, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <p>Enter your password if prompted.</p>
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ipv6 mld snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# ipv6 mld snooping vlan 1 mrouter interface gigabitethernet 0/2</pre>	<p>Specifies the multicast router VLAN ID, and specify the interface to the multicast router.</p> <ul style="list-style-type: none"> • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 48.
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 5	<p>show ipv6 mld snooping mrouter [vlan <i>vlan-id</i>]</p> <p>Example:</p> <pre>Switch# show ipv6 mld snooping mrouter vlan 1</pre>	<p>Verifies that IPv6 MLD snooping is enabled on the VLAN interface.</p>

Enabling MLD Immediate Leave

To enable MLDv1 immediate leave, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 mld snooping vlan <i>vlan-id</i> immediate-leave Example: Switch(config)# ipv6 mld snooping vlan 1 immediate-leave	Enables MLD Immediate Leave on the VLAN interface.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show ipv6 mld snooping vlan <i>vlan-id</i> Example: Switch# show ipv6 mld snooping vlan 1	Verifies that Immediate Leave is enabled on the VLAN interface.

Configuring MLD Snooping Queries

To configure MLD snooping query characteristics for the switch or for a VLAN, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	ipv6 mld snooping robustness-variable <i>value</i> Example: Switch(config)# <code>ipv6 mld snooping robustness-variable 3</code>	(Optional) Sets the number of queries that are sent before switch will deletes a listener (port) that does not respond to a general query. The range is 1 to 3; the default is 2.
Step 4	ipv6 mld snooping vlan <i>vlan-id</i> robustness-variable <i>value</i> Example: Switch(config)# <code>ipv6 mld snooping vlan 1 robustness-variable 3</code>	(Optional) Sets the robustness variable on a VLAN basis, which determines the number of general queries that MLD snooping sends before aging out a multicast address when there is no MLD report response. The range is 1 to 3; the default is 0. When set to 0, the number used is the global robustness variable value.
Step 5	ipv6 mld snooping last-listener-query-count <i>count</i> Example: Switch(config)# <code>ipv6 mld snooping last-listener-query-count 7</code>	(Optional) Sets the number of MASQs that the switch sends before aging out an MLD client. The range is 1 to 7; the default is 2. The queries are sent 1 second apart.
Step 6	ipv6 mld snooping vlan <i>vlan-id</i> last-listener-query-count <i>count</i> Example: Switch(config)# <code>ipv6 mld snooping vlan 1 last-listener-query-count 7</code>	(Optional) Sets the last-listener query count on a VLAN basis. This value overrides the value configured globally. The range is 1 to 7; the default is 0. When set to 0, the global count value is used. Queries are sent 1 second apart.
Step 7	ipv6 mld snooping last-listener-query-interval <i>interval</i> Example: Switch(config)# <code>ipv6 mld snooping last-listener-query-interval 2000</code>	(Optional) Sets the maximum response time that the switch waits after sending out a MASQ before deleting a port from the multicast group. The range is 100 to 32,768 thousands of a second. The default is 1000 (1 second).
Step 8	ipv6 mld snooping vlan <i>vlan-id</i> last-listener-query-interval <i>interval</i> Example: Switch(config)# <code>ipv6 mld snooping vlan 1 last-listener-query-interval 2000</code>	(Optional) Sets the last-listener query interval on a VLAN basis. This value overrides the value configured globally. The range is 0 to 32,768 thousands of a second. The default is 0. When set to 0, the global last-listener query interval is used.
Step 9	ipv6 mld snooping tcn query solicit Example: Switch(config)# <code>ipv6 mld snooping tcn query solicit</code>	(Optional) Enables topology change notification (TCN) solicitation, which means that VLANs flood all IPv6 multicast traffic for the configured number of queries before sending multicast data to only those ports requesting to receive it. The default is for TCN to be disabled.

	Command or Action	Purpose
Step 10	ipv6 mld snooping tcn flood query count <i>count</i> Example: Switch(config)# ipv6 mld snooping tcn flood query count 5	(Optional) When TCN is enabled, specifies the number of TCN queries to be sent. The range is from 1 to 10; the default is 2.
Step 11	end	Returns to privileged EXEC mode.
Step 12	show ipv6 mld snooping querier [vlan <i>vlan-id</i>] Example: Switch(config)# show ipv6 mld snooping querier vlan 1	(Optional) Verifies that the MLD snooping querier information for the switch or for the VLAN.

Disabling MLD Listener Message Suppression

MLD snooping listener message suppression is enabled by default. When it is enabled, the switch forwards only one MLD report per multicast router query. When message suppression is disabled, multiple MLD reports could be forwarded to the multicast routers.

To disable MLD listener message suppression, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enter global configuration mode.
Step 3	no ipv6 mld snooping listener-message-suppression Example: Switch(config)# no ipv6 mld snooping listener-message-suppression	Disable MLD message suppression.
Step 4	end Example: Switch(config)# end	Return to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show ipv6 mld snooping Example: Switch# <code>show ipv6 mld snooping</code>	Verify that IPv6 MLD snooping report suppression is disabled.

Displaying MLD Snooping Information

You can display MLD snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display IPv6 group address multicast entries for a VLAN configured for MLD snooping.

Table 34: Commands for Displaying MLD Snooping Information

Command	Purpose
show ipv6 mld snooping [vlan <i>vlan-id</i>]	Displays the MLD snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping mrouter [vlan <i>vlan-id</i>]	Displays information on dynamically learned and manually configured multicast router interfaces. When you enable MLD snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enters vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping querier [vlan <i>vlan-id</i>]	Displays information about the IPv6 address and incoming port for the most-recently received MLD query messages in the VLAN. (Optional) Enters vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping address [vlan <i>vlan-id</i>] [count dynamic user]	Displays all IPv6 multicast address information or specific IPv6 multicast address information for the switch or a VLAN. <ul style="list-style-type: none"> • Enters count to show the group count on the switch or in a VLAN. • Enters dynamic to display MLD snooping learned group information for the switch or for a VLAN. • Enters user to display MLD snooping user-configured group information for the switch or for a VLAN.
show ipv6 mld snooping address vlan <i>vlan-id</i> [<i>ipv6-multicast-address</i>]	Displays MLD snooping for the specified VLAN and IPv6 multicast address.

Configuration Examples for Configuring MLD Snooping

Configuring a Static Multicast Group: Example

This example shows how to statically configure an IPv6 multicast group:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 2 static 3333.0000.1111 interface gigabitethernet1/0/1
Switch(config)# end
```

Configuring a Multicast Router Port: Example

This example shows how to add a multicast router port to VLAN 200:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet
                0/2
Switch(config)# exit
```

Enabling MLD Immediate Leave: Example

This example shows how to enable MLD Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 130 immediate-leave
Switch(config)# exit
```

Configuring MLD Snooping Queries: Example

This example shows how to set the MLD snooping global robustness variable to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping robustness-variable 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query count for a VLAN to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query interval (maximum response time) to 2000 (2 seconds):

```
Switch# configure terminal  
Switch(config)# ipv6 mld snooping last-listener-query-interval 2000  
Switch(config)# exit
```




CHAPTER 21

Configuring IPv6 Unicast Routing

- [Finding Feature Information, on page 303](#)
- [Information About Configuring IPv6 Unicast Routing, on page 303](#)
- [Configuring DHCP for IPv6 Address Assignment, on page 350](#)
- [Configuration Examples for IPv6 Unicast Routing, on page 354](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Information About Configuring IPv6 Unicast Routing

This chapter describes how to configure IPv6 unicast routing on the switch.

Understanding IPv6

IPv4 users can move to IPv6 and receive services such as end-to-end security, quality of service (QoS), and globally unique addresses. The IPv6 address space reduces the need for private addresses and Network Address Translation (NAT) processing by border routers at network edges.

For information about how Cisco Systems implements IPv6, go to:

http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

For information about IPv6 and other features in this chapter

- See the *Cisco IOS IPv6 Configuration Library*.
- Use the Search field on Cisco.com to locate the Cisco IOS software documentation. For example, if you want information about static routes, you can enter *Implementing Static Routes for IPv6* in the search field to learn about static routes.

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, or anycast addresses.

The IPv6 128-bit addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons in the format: n:n:n:n:n:n:n:n. This is an example of an IPv6 address:

```
2031:0000:130F:0000:0000:09C0:080F:130B
```

For easier implementation, leading zeros in each field are optional. This is the same address without leading zeros:

```
2031:0:130F:0:0:9C0:80F:130B
```

You can also use two colons (::) to represent successive hexadecimal fields of zeros, but you can use this short version only once in each address:

```
2031:0:130F::09C0:080F:130B
```

For more information about IPv6 address formats, address types, and the IPv6 packet header, see the http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xr-3e/ipv6-xr-3e-book.html of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

In the "Implementing Addressing and Basic Connectivity" chapter, these sections apply to the Catalyst 2960, 2960-S, 2960-C, 2960-X, 2960-CX and 3560-CX switches:

- IPv6 Address Formats
- IPv6 Address Type: Multicast
- IPv6 Address Output Display
- Simplified IPv6 Packet Header

Supported IPv6 Unicast Routing Features

These sections describe the IPv6 protocol features supported by the switch:

128-Bit Wide Unicast Addresses

The switch supports aggregatable global unicast addresses and link-local unicast addresses. It does not support site-local unicast addresses.

- Aggregatable global unicast addresses are IPv6 addresses from the aggregatable global unicast prefix. The address structure enables strict aggregation of routing prefixes and limits the number of routing table entries in the global routing table. These addresses are used on links that are aggregated through organizations and eventually to the Internet service provider.

These addresses are defined by a global routing prefix, a subnet ID, and an interface ID. Current global unicast address allocation uses the range of addresses that start with binary value 001 (2000::/3). Addresses with a prefix of 2000::/3(001) through E000::/3(111) must have 64-bit interface identifiers in the extended unique identifier (EUI)-64 format.

- Link local unicast addresses can be automatically configured on any interface by using the link-local prefix FE80::/10(1111 1110 10) and the interface identifier in the modified EUI format. Link-local addresses are used in the neighbor discovery protocol (NDP) and the stateless autoconfiguration process. Nodes on a local link use link-local addresses and do not require globally unique addresses to communicate. IPv6 routers do not forward packets with link-local source or destination addresses to other links.

For more information, see the section about IPv6 unicast addresses in the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DNS for IPv6

IPv6 supports Domain Name System (DNS) record types in the DNS name-to-address and address-to-name lookup processes. The DNS AAAA resource record types support IPv6 addresses and are equivalent to an A address record in IPv4. The switch supports DNS resolution for IPv4 and IPv6.

Path MTU Discovery for IPv6 Unicast

The switch supports advertising the system maximum transmission unit (MTU) to IPv6 nodes and path MTU discovery. Path MTU discovery allows a host to dynamically discover and adjust to differences in the MTU size of every link along a given data path. In IPv6, if a link along the path is not large enough to accommodate the packet size, the source of the packet handles the fragmentation.

ICMPv6

The Internet Control Message Protocol (ICMP) in IPv6 generates error messages, such as ICMP destination unreachable messages, to report errors during processing and other diagnostic functions. In IPv6, ICMP packets are also used in the neighbor discovery protocol and path MTU discovery.

Neighbor Discovery

The switch supports NDP for IPv6, a protocol running on top of ICMPv6, and static neighbor entries for IPv6 stations that do not support NDP. The IPv6 neighbor discovery process uses ICMP messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), to verify the reachability of the neighbor, and to keep track of neighboring routers.

The switch supports ICMPv6 redirect for routes with mask lengths less than 64 bits. ICMP redirect is not supported for host routes or for summarized routes with mask lengths greater than 64 bits.

Neighbor discovery throttling ensures that the switch CPU is not unnecessarily burdened while it is in the process of obtaining the next hop forwarding information to route an IPv6 packet. The switch drops any additional IPv6 packets whose next hop is the same neighbor that the switch is actively trying to resolve. This drop avoids further load on the CPU.

Default Router Preference

The switch supports IPv6 default router preference (DRP), an extension in router advertisement messages. DRP improves the ability of a host to select an appropriate router, especially when the host is multihomed and the routers are on different links. The switch does not support the Route Information Option in RFC 4191.

An IPv6 host maintains a default router list from which it selects a router for traffic to offlink destinations. The selected router for a destination is then cached in the destination cache. NDP for IPv6 specifies that routers that are reachable or probably reachable are preferred over routers whose reachability is unknown or suspect. For reachable or probably reachable routers, NDP can either select the same router every time or cycle through the router list. By using DRP, you can configure an IPv6 host to prefer one router over another, provided both are reachable or probably reachable.

For configuring DRP for IPv6, see the *Configuring Default Router Preference* section.

For more information about DRP for IPv6, see the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Stateless Autoconfiguration and Duplicate Address Detection

The switch uses stateless autoconfiguration to manage link, subnet, and site addressing changes, such as management of host and mobile IP addresses. A host autonomously configures its own link-local address, and booting nodes send router solicitations to request router advertisements for configuring interfaces.

For more information about autoconfiguration and duplicate address detection, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Applications

The switch has IPv6 support for these applications:

- Ping, traceroute, and Telnet
- Secure Shell (SSH) over an IPv6 transport
- HTTP server access over IPv6 transport
- DNS resolver for AAAA over IPv4 transport
- Cisco Discovery Protocol (CDP) support for IPv6 addresses

For more information about managing these applications, see the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DHCP for IPv6 Address Assignment

DHCPv6 enables DHCP servers to pass configuration parameters, such as IPv6 network addresses, to IPv6 clients. The address assignment feature manages non-duplicate address assignment in the correct prefix based on the network where the host is connected. Assigned addresses can be from one or multiple prefix pools. Additional options, such as default domain and DNS name-server address, can be passed back to the client. Address pools can be assigned for use on a specific interface, on multiple interfaces, or the server can automatically find the appropriate pool.

For configuring DHCP for IPv6, see the *Configuring DHCP for IPv6 Address Assignment* section.

For more information about configuring the DHCPv6 client, server, or relay agent functions, see the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Static Routes for IPv6

Static routes are manually configured and define an explicit route between two networking devices. Static routes are useful for smaller networks with only one path to an outside network or to provide security for certain types of traffic in a larger network.

Configuring Static Routing for IPv6 (CLI)

For configuring static routes for IPv6, see the *Configuring Static Routing for IPv6* section.

For more information about static routes, see the “Implementing Static Routes for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

RIP for IPv6

Routing Information Protocol (RIP) for IPv6 is a distance-vector protocol that uses hop count as a routing metric. It includes support for IPv6 addresses and prefixes and the all-RIP-routers multicast group address FF02::9 as the destination address for RIP update messages.

For configuring RIP for IPv6, see the *Configuring RIP for IPv6* section.

For more information about RIP for IPv6, see the “Implementing RIP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

OSPF for IPv6

The switch running the feature set supports Open Shortest Path First (OSPF) for IPv6, a link-state protocol for IP. For more information, see *Cisco IOS IPv6 Configuration Library* on Cisco.com.

OSPFv3 Graceful Restart

OSPFv3 feature allows nonstop data forwarding along known routes while the OSPFv3 routing protocol information is restored. A switch uses graceful restart either in restart mode (for a graceful-restart-capable switch) or in helper mode (for a graceful-restart-aware switch).

To use the graceful restart function, a switch must be in high-availability stateful switchover (SSO) mode (dual route processor). A switch capable of graceful restart uses it when these failures occur:

- A route processor failure that results in changeover to the standby route processor
- A planned route processor changeover to the standby route processor

The graceful restart feature requires that neighboring switches be graceful-restart aware.

For more information, see the Implementing OSPF for IPv6 chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Fast Convergence: LSA and SPF Throttling

The OSPFv3 link-state advertisements (LSA) and shortest path first (SPF) throttling feature provides a dynamic method to slow down link-state advertisement updates in OSPFv3 during times of network instability. This feature also allows faster OSPFv3 convergence by providing LSA rate limiting in milliseconds.

OSPFv3 previously used static timers for rate-limiting SPF calculation and LSA generation. Although these timers are configurable, the values are specified in seconds, which poses a limitation on OSPFv3 convergence. LSA and SPF throttling achieves subsecond convergence by providing a more sophisticated SPF and LSA rate-limiting method can react quickly to changes and also provide stability and protection during prolonged periods of instability.

Authentication Support with IPsec

To ensure that OSPF for IPv6 (OSPFv3) packets are not altered and resent to the switch, OSPFv3 packets must be authenticated. OSPFv3 uses the IPsec secure socket API to add authentication to OSPFv3 packets. This API has been extended to provide support for IPv6.

OSPFv3 requires the use of IPsec to enable authentication. Crypto images are required to use authentication, because only crypto images include the IPsec API needed for use with OSPFv3.

Configuring HSRP for IPv6

HSRP provides routing redundancy for routing IPv6 traffic not dependent on the availability of any single router. IPv6 hosts learn of available routers through IPv6 neighbor discovery router advertisement messages. These messages are multicast periodically or are solicited by hosts.

An HSRP IPv6 group has a virtual MAC address that is derived from the HSRP group number and a virtual IPv6 link-local address that is, by default, derived from the HSRP virtual MAC address. Periodic messages

are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active. These messages stop after a final one is sent when the group leaves the active state.



Note When configuring HSRP for IPv6, you must enable HSRP version 2 (HSRPv2) on the interface.

EIGRP IPv6

Switches running the IP services feature set support the Enhanced Interior Gateway Routing Protocol (EIGRP) for IPv6. It is configured on the interfaces on which it runs and does not require a global IPv6 address.



Note Switches running the IP base feature set do not support any IPv6 EIGRP features, including IPv6 EIGRP stub routing.

Before running, an instance of EIGRP IPv6 requires an implicit or explicit router ID. An implicit router ID is derived from a local IPv4 address, so any IPv4 node always has an available router ID. However, EIGRP IPv6 might be running in a network with only IPv6 nodes and therefore might not have an available IPv4 router ID.

For more information about EIGRP for IPv6, see the “Implementing EIGRP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

SNMP and Syslog Over IPv6

To support both IPv4 and IPv6, IPv6 network management requires both IPv6 and IPv4 transports. Syslog over IPv6 supports address data types for these transports.

Simple Network Management Protocol (SNMP) and syslog over IPv6 provide these features:

- Support for both IPv4 and IPv6
- IPv6 transport for SNMP and to modify the SNMP agent to support traps for an IPv6 host
- SNMP- and syslog-related MIBs to support IPv6 addressing
- Configuration of IPv6 hosts as trap receivers

For support over IPv6, SNMP modifies the existing IP transport mapping to simultaneously support IPv4 and IPv6. These SNMP actions support IPv6 transport management:

- Opens User Datagram Protocol (UDP) SNMP socket with default settings
- Provides a new transport mechanism called *SR_IPV6_TRANSPORT*
- Sends SNMP notifications over IPv6 transport
- Supports SNMP-named access lists for IPv6 transport
- Supports SNMP proxy forwarding using IPv6 transport
- Verifies SNMP Manager feature works with IPv6 transport

For information on SNMP over IPv6, including configuration procedures, see the “Managing Cisco IOS Applications over IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

For information about syslog over IPv6, including configuration procedures, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

HTTP(S) Over IPv6

The HTTP client sends requests to both IPv4 and IPv6 HTTP servers, which respond to requests from both IPv4 and IPv6 HTTP clients. URLs with literal IPv6 addresses must be specified in hexadecimal using 16-bit values between colons.

The accept socket call chooses an IPv4 or IPv6 address family. The accept socket is either an IPv4 or IPv6 socket. The listening socket continues to listen for both IPv4 and IPv6 signals that indicate a connection. The IPv6 listening socket is bound to an IPv6 wildcard address.

The underlying TCP/IP stack supports a dual-stack environment. HTTP relies on the TCP/IP stack and the sockets for processing network-layer interactions.

Basic network connectivity (**ping**) must exist between the client and the server hosts before HTTP connections can be made.

For more information, see the “Managing Cisco IOS Applications over IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Unsupported IPv6 Unicast Routing Features

The switch does not support these IPv6 features:

- IPv6 packets destined to site-local addresses
- Tunneling protocols, such as IPv4-to-IPv6 or IPv6-to-IPv4
- The switch as a tunnel endpoint supporting IPv4-to-IPv6 or IPv6-to-IPv4 tunneling protocols

IPv6 Feature Limitations

Because IPv6 is implemented in switch hardware, some limitations occur due to the IPv6 compressed addresses in the hardware memory. These hardware limitations result in some loss of functionality and limits some features.

These are feature limitations.

- The switch cannot forward SNAP-encapsulated IPv6 packets in hardware. They are forwarded in software.
- The switch cannot apply QoS classification on source-routed IPv6 packets in hardware.

Configuring IPv6

Default IPv6 Configuration

Table 35: Default IPv6 Configuration

Feature	Default Setting
SDM template	Advance desktop. Default is advanced template Default

Feature	Default Setting
IPv6 addresses	None configured

Configuring IPv6 Addressing and Enabling IPv6 Routing

This section describes how to assign IPv6 addresses to individual Layer 3 interfaces and to globally forward IPv6 traffic on the switch.

Before configuring IPv6 on the switch, consider these guidelines:

- Be sure to select a dual IPv4 and IPv6 SDM template.
- In the **ipv6 address** interface configuration command, you must enter the *ipv6-address* and *ipv6-prefix* variables with the address specified in hexadecimal using 16-bit values between colons. The *prefix-length* variable (preceded by a slash [/]) is a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address).

To forward IPv6 traffic on an interface, you must configure a global IPv6 address on that interface. Configuring an IPv6 address on an interface automatically configures a link-local address and activates IPv6 for the interface. The configured interface automatically joins these required multicast groups for that link:

- solicited-node multicast group FF02:0:0:0:0:1:ff00::/104 for each unicast address assigned to the interface (this address is used in the neighbor discovery process.)
- all-nodes link-local multicast group FF02::1
- all-routers link-local multicast group FF02::2

For more information about configuring IPv6 routing, see the “Implementing Addressing and Basic Connectivity for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Beginning in privileged EXEC mode, follow these steps to assign an IPv6 address to a Layer 3 interface and enable IPv6 routing:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode after the switch reloads.
Step 2	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure. The interface can be a physical interface, a switch virtual interface (SVI), or a Layer 3 EtherChannel.
Step 3	no switchport Example:	Removes the interface from Layer 2 configuration mode (if it is a physical interface).

	Command or Action	Purpose
	Switch(config-if) # no switchport	
Step 4	<p>Use one of the following:</p> <ul style="list-style-type: none"> • ipv6 address <i>ipv6-prefix/prefix length eui-64</i> • ipv6 address <i>ipv6-address/prefix length</i> • ipv6 address <i>ipv6-address link-local</i> • ipv6 enable <p>Example:</p> <pre>Switch(config-if) # ipv6 address 2001:0DB8:c18:1::/64 eui 64 Switch(config-if) # ipv6 address 2001:0DB8:c18:1::/64 Switch(config-if) # ipv6 address 2001:0DB8:c18:1:: link-local Switch(config-if) # ipv6 enable</pre>	<ul style="list-style-type: none"> • Specifies a global IPv6 address with an extended unique identifier (EUI) in the low-order 64 bits of the IPv6 address. Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. This enables IPv6 processing on the interface. • Manually configures an IPv6 address on the interface. • Specifies a link-local address on the interface to be used instead of the link-local address that is automatically configured when IPv6 is enabled on the interface. This command enables IPv6 processing on the interface. • Automatically configures an IPv6 link-local address on the interface, and enables the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.
Step 5	<p>exit</p> <p>Example:</p> <pre>Switch(config-if) # exit</pre>	Returns to global configuration mode.
Step 6	<p>ip routing</p> <p>Example:</p> <pre>Switch(config) # ip routing</pre>	Enables IP routing on the switch.
Step 7	<p>ipv6 unicast-routing</p> <p>Example:</p> <pre>Switch(config) # ipv6 unicast-routing</pre>	Enables forwarding of IPv6 unicast data packets.
Step 8	<p>end</p> <p>Example:</p> <pre>Switch(config) # end</pre>	Returns to privileged EXEC mode.
Step 9	<p>show ipv6 interface interface-id</p> <p>Example:</p>	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show ipv6 interface gigabitethernet 1/0/1</code>	
Step 10	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring First Hop Security in IPv6

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.
- QoS should be enabled on the switch before configuring CoPP policies using `mls qos` command.

Restrictions for First Hop Security in IPv6

- The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):
 - A physical port with an FHS policy attached cannot join an EtherChannel group.
 - An FHS policy cannot be attached to a physical port when it is a member of an EtherChannel group.
- By default, a snooping policy has a security-level of guard. When such a snooping policy is configured on an access switch, external IPv6 Router Advertisement (RA) or Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server packets are blocked, even though the uplink port facing the router or DHCP server/relay is configured as a trusted port. To allow IPv6 RA or DHCPv6 server messages, do the following:
 - Apply an IPv6 RA-guard policy (for RA) or IPv6 DHCP-guard policy (for DHCP server messages) on the uplink port.
 - Configure a snooping policy with a lower security-level, for example `glean` or `inspect`. However; configuring a lower security level is not recommended with such a snooping policy, because benefits of First Hop security features are not effective.
- The following restrictions apply for CoPP policies with IPv6 SISF-based device tracking policies due to limitation reported in [CSCvk32439](#):
 - CoPP policies are required to limit IPv6 NDP traffic when IPv6 SISF policies are configured on the switch.
 - After NDP CoPP policies are configured, limited traffic hits CPU. To accommodate the total end points connected, the number of NDP CoPP policies should be slightly more than the number of users connected to each switch in a stack. If you configure NDP CoPP policies less than the number of end points connected to the switch, the IP allocation to the end point is delayed but is not ignored completely.



Note For example, if a stack of 5 switches has approximately 300 users, the NDP CoPP policies should be more than 300.

- The DHCPv6 (server-to-client and client-to-server) CoPP policies are required only if Lightweight DHCPv6 Relay Agent (LDRA) is configured under IPv6 SISF-based device tracking policies on the switch.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.
- IPv6 FHS Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.
- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

This feature mitigates some of the inherent vulnerabilities of the ND mechanism, such as attacks on DAD, address resolution, router discovery, and the neighbor cache.

- IPv6 Router Advertisement Guard—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the Layer 2 device with the information found in the received RA frame. Once the Layer 2 device has validated the content of the RA frame and router redirect frame against the configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.
- IPv6 DHCP Guard—The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.
- IPv6 Source Guard—Like IPv4 Source Guard, IPv6 Source Guard validates the source address or prefix to prevent source address spoofing.

A source guard programs the hardware to allow or deny traffic based on source or destination addresses. It deals exclusively with data packet traffic.

To debug source-guard packets, use the `debug ipv6 snooping source-guard` privileged EXEC command.

The following restrictions apply:

- An FHS policy cannot be attached to a physical port when it is a member of an EtherChannel group.
- When IPv6 source guard is enabled on a switch port, NDP or DHCP snooping must be enabled on the interface to which the switch port belongs. Otherwise, all data traffic from this port will be blocked.
- An IPv6 source guard policy cannot be attached to a VLAN. It is supported only at the interface level.
- When you configure IPv4 and IPv6 source guard together on an interface, it is recommended to use **ip verify source mac-check** instead of **ip verify source**. IPv4 connectivity on a given port might break due to two different filtering rules set — one for IPv4 (IP-filter) and the other for IPv6 (IP-MAC filter).
- You cannot use IPv6 Source Guard and Prefix Guard together. When you attach the policy to an interface, it should be "validate address" or "validate prefix" but not both.
- PVLAN and Source/Prefix Guard cannot be applied together.

For more information on IPv6 Source Guard, see the [IPv6 Source Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Prefix Guard—The IPv6 prefix guard feature works within the IPv6 source guard feature, to enable the device to deny traffic originated from non-topologically correct addresses. IPv6 prefix guard is often used when IPv6 prefixes are delegated to devices (for example, home gateways) using DHCP prefix delegation. The feature discovers ranges of addresses assigned to the link and blocks any traffic sourced with an address outside this range.

For more information on IPv6 Prefix Guard, see the [IPv6 Prefix Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Destination Guard—The IPv6 destination guard feature works with IPv6 neighbor discovery to ensure that the device performs address resolution only for those addresses that are known to be active on the link. It relies on the address glean functionality to populate all destinations active on the link into the binding table and then blocks resolutions before they happen when the destination is not found in the binding table.

For more information about IPv6 Destination Guard, see the [IPv6 Destination Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Neighbor Discovery Multicast Suppress—The IPv6 Neighbor Discovery multicast suppress feature is an IPv6 snooping feature that runs on a switch or a wireless controller and is used to reduce the amount of control traffic necessary for proper link operations.
- DHCPv6 Relay—Lightweight DHCPv6 Relay Agent—The DHCPv6 Relay—Lightweight DHCPv6 Relay Agent feature allows relay agent information to be inserted by an access node that performs a link-layer bridging (non-routing) function. Lightweight DHCPv6 Relay Agent (LDRA) functionality can be implemented in existing access nodes, such as DSL access multiplexers (DSLAMs) and Ethernet switches, that do not support IPv6 control or routing functions. LDRA is used to insert relay-agent options

in DHCP version 6 (DHCPv6) message exchanges primarily to identify client-facing interfaces. LDRA functionality can be enabled on an interface and on a VLAN.



Note If an LDRA device is directly connected to a client, the interface must have the pool configuration to fetch the specific subnet or link information at the server side. In this case, if the LDRA device is present in different subnets or links, the server may not be able to fetch the correct subnet. You can now configure the pool name in the interface so as to choose the proper subnet or link for the client.

For more information about DHCPv6 Relay, See the [DHCPv6 Relay—Lightweight DHCPv6 Relay Agent](#) section of the IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15.1SG.

How to configure an IPv6 Snooping Policy

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **IPv6 snooping policy *policy -name***
4. **[data-glean | default | device-role [node|switch] | limit {address-countvalue} | no | protocol [all | nodhcp | ndp] | security-level [glean | guard | inspect] | tracking [disable | enable] | trusted-port }**
5. **exit**
6. **show ipv6 snooping policy *policy-name***

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	IPv6 snooping policy <i>policy -name</i>	Creates a snooping policy in global configuration mode.
Step 4	[data-glean default device-role [node switch] limit {address-countvalue} no protocol [all nodhcp ndp] security-level [glean guard inspect] tracking [disable enable] trusted-port }	Enables data address gleaning, validates messages against various criteria, specifies the security level for messages. • (Optional) data-glean —Enables data address gleaning. This option is disabled by default. • (Optional) default —Sets all default options. • (Optional) device-role [node switch] —Qualifies the role of the device attached to the port.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) limit {address-count <i>value</i>}—Limits the number of addresses allowed per target. • (Optional) no—Negates a command or set its defaults. • (Optional) protocol [all dhcp ndp]—Specifies which protocol should be redirected to the snooping feature for analysis. The default, is all. To change the default, use the no protocol command. • (Optional) security-level [glean guard inspect]—Specifies the level of security enforced by the feature. <ul style="list-style-type: none"> • glean—Gleans addresses from messages and populates the binding table without any verification. • guard—Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option. • inspect—Gleans addresses, validates messages for consistency and conformance, and enforces address ownership. • (Optional) tracking [disable enable]—Overrides the default tracking behavior and specifies a tracking option. • (Optional) trusted-port—Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is also given preference in case of a collision while making an entry in the table.
Step 5	exit	Exits the snooping policy configuration mode.
Step 6	show ipv6 snooping policy <i>policy-name</i>	Displays the snooping policy configuration.

How to Attach an IPv6 Snooping Policy to an Interface or VLAN

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
 - **interface** *type number*
 - **switchport**

- **ipv6 snooping** [**attach-policy** *policy_name*]

OR

- **vlan configuration** *vlan list*
- **ipv6 snooping attach-policy** *policy-name*

4. **show ipv6 snooping policy** *policy-name*
5. **show ipv6 neighbors binding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	Perform one of the following tasks: <ul style="list-style-type: none"> • interface <i>type number</i> • switchport • ipv6 snooping [attach-policy <i>policy_name</i>] OR <ul style="list-style-type: none"> • vlan configuration <i>vlan list</i> • ipv6 snooping attach-policy <i>policy-name</i> 	Specifies an interface type and number, and enters the interface configuration mode. <p>Note <i>type</i> can be physical interface or ether-channel.</p> Configures the interface as a Layer 2 port. Attaches the snooping policy (where data gleaning is enabled) to an interface. Specifies the port and the policy that is attached to the port. <p>Note If you have enabled data-glean on a snooping policy, you must attach it to an interface and not a VLAN.</p>
Step 4	show ipv6 snooping policy <i>policy-name</i>	Displays the snooping policy configuration.
Step 5	show ipv6 neighbors binding	Displays the binding table entries populated by the snooping policy.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on a Device

To attach an IPV6 Neighbor Discovery Multicast Suppress policy on a device, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 nd suppress policy** *policy-name*
4. **mode dad-proxy**
5. **mode full-proxy**

6. mode mc-proxy

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	ipv6 nd suppress policy <i>policy-name</i>	Defines the Neighbor Discovery suppress policy name and enters Neighbor Discovery suppress policy configuration mode.
Step 4	mode dad-proxy	Enables Neighbor Discovery suppress in IPv6 DAD proxy mode.
Step 5	mode full-proxy	Enables Neighbor Discovery suppress to proxy multicast and unicast Neighbor Solicitation messages.
Step 6	mode mc-proxy	Enables Neighbor Discovery suppress to proxy multicast Neighbor Solicitation messages.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on an Interface

To attach an IPv6 Neighbor Discovery Multicast Suppress policy on an interface, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
 - **interface** *type number*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1, vlan2, vlan3...*]]]
 - OR
 - **vlan configuration** *vlan-id*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1, vlan2, vlan3...*]]]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	Perform one of the following tasks: <ul style="list-style-type: none"> • interface <i>type number</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] OR • vlan configuration <i>vlan-id</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] 	Specifies an interface type and number, and places the device in interface configuration mode. Attaches the IPv6 Neighbor Discovery Multicast Policy to an interface or a VLAN.
Step 4	exit	Exits the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy to a Layer 2 EtherChannel Interface

To attach an IPv6 Neighbor Discovery Multicast Suppress policy on an EtherChannel interface, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
 - **interface port-channel** *port-channel-number*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1*, *vlan2*, *vlan3*...]]]
 - OR
 - **vlan configuration** *vlan-id*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1*, *vlan2*, *vlan3*...]]]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	Perform one of the following tasks: <ul style="list-style-type: none"> • interface port-channel <i>port-channel-number</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] OR <ul style="list-style-type: none"> • vlan configuration <i>vlan-id</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] 	Specifies an interface type and port number and places the switch in the port channel configuration mode. Attaches the IPv6 Neighbor Discovery Multicast Policy to an interface or a VLAN.
Step 4	exit	Exits the interface configuration mode.

How to Configure an IPv6 DHCP Guard Policy

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 dhcp guard policy** *policy-name*
4. [**default** | **device-role** [**client** | **server**] | **no** | **exit** | **trusted-port**]
5. **exit**
6. Perform one of the following tasks:
 - **interface** *type number*
 - **ipv6 dhcp guard attach-policy** *policy-name*
 OR
 - **vlan configuration** *vlan-id*
 - **ipv6 dhcp guard attach-policy** *policy-name*
7. **show ipv6 dhcp guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	ipv6 dhcp guard policy <i>policy-name</i>	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.
Step 4	[default device-role [client server] no exit trusted-port]	(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is client . <ul style="list-style-type: none"> • client—Default value, specifies that the attached device is a client. Server messages are dropped on this port. • server—Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port. (Optional) trusted-port —Sets the port to a trusted mode. No further policing takes place on the port. Note If you configure a trusted port then the device-role option is not available.
Step 5	exit	Exits the DHCP guard policy global configuration mode.
Step 6	Perform one of the following tasks: <ul style="list-style-type: none"> • interface <i>type number</i> • ipv6 dhcp guard attach-policy <i>policy-name</i> OR <ul style="list-style-type: none"> • vlan configuration <i>vlan-id</i> • ipv6 dhcp guard attach-policy <i>policy-name</i> 	Specifies an interface type and number and enters the interface configuration mode. Attaches the DHCP guard policy to an interface or VLAN.
Step 7	show ipv6 dhcp guard policy <i>policy_name</i>	Displays the DHCP guard policy configuration.

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
  permit host FE80::A8BB:CCFF:FE01:F700 any
```

```

ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy poll
  device-role server
  match server access-list acl1
  match reply prefix-list abc
  preference min 0
  preference max 255
  trusted-port
interface GigabitEthernet 0/2/0
  switchport
  ipv6 dhcp guard attach-policy poll vlan add 1
  vlan configuration 1
  ipv6 dhcp guard attach-policy poll
show ipv6 dhcp guard policy poll

```

How to Configure IPv6 Source Guard

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 source-guard policy *policy_name***
4. **[deny global-autoconf] [permit link-local] [default{...}] [exit] [no{...}]**
5. **ipv6 source-guard [attach-policy*policy-name*]**
6. **exit**
7. **show ipv6 source-guard policy*policy_name***

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	ipv6 source-guard policy <i>policy_name</i>	Specifies the IPv6 Source Guard policy name and enters IPv6 Source Guard policy configuration mode.
Step 4	[deny global-autoconf] [permit link-local] [default{...}] [exit] [no{...}]	Defines the IPv6 Source Guard policy. <ul style="list-style-type: none"> • deny global-autoconf—Denies data traffic from auto-configured global addresses. This is useful when all global addresses on a link are DHCP-assigned and the administrator wants to block hosts with self-configured addresses to send traffic. • permit link-local—Allows all data traffic that is sourced by a link-local address.
Step 5	ipv6 source-guard [attach-policy<i>policy-name</i>]	Specifies the policy name.

	Command or Action	Purpose
		(Optional) attach-policy <i>policy-name</i> —Filters based on the policy name
Step 6	exit	Exits the source guard policy configuration mode.
Step 7	show ipv6 source-guard policy <i>policy_name</i>	Shows the policy configuration and all the interfaces where the policy is applied.

Configuring Default Router Preference

Router advertisement messages are sent with the default router preference (DRP) configured by the **ipv6 nd router-preference** interface configuration command. If no DRP is configured, RAs are sent with a medium preference.

A DRP is useful when two routers on a link might provide equivalent, but not equal-cost routing, and policy might dictate that hosts should prefer one of the routers.

For more information about configuring DRP for IPv6, see the “Implementing IPv6 Addresses and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

To configure a DRP for a router on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode and identifies the Layer 3 interface on which you want to specify the DRP.
Step 4	ipv6 nd router-preference { high medium low } Example: Switch(config-if)# ipv6 nd router-preference medium	Specifies a DRP for the router on the switch interface.

	Command or Action	Purpose
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show ipv6 interface Example: Switch# show ipv6 interface	Verifies the configuration.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring IPv6 ICMP Rate Limiting

ICMP rate limiting is enabled by default with a default interval between error messages of 100 milliseconds and a bucket size (maximum number of tokens to be stored in a bucket) of 10.

To change the ICMP rate-limiting parameters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 icmp error-interval interval [bucketsize] Example: Switch(config)# ipv6 icmp error-interval 50 20	Configures the interval and bucket size for IPv6 ICMP error messages: <ul style="list-style-type: none"> <i>interval</i>—The interval (in milliseconds) between tokens being added to the bucket. The range is from 0 to 2147483647 milliseconds. <i>bucketsize</i>—(Optional) The maximum number of tokens stored in the bucket. The range is from 1 to 200.

	Command or Action	Purpose
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show ipv6 interface [<i>interface-id</i>] Example: <pre>Switch# show ipv6 interface gigabitethernet0/1</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Cisco Express Forwarding and distributed Cisco Express Forwarding for IPv6

Cisco Express Forwarding is a Layer 3 IP switching technology to improve network performance. Cisco Express Forwarding implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. It is less CPU-intensive than fast-switching route-caching, allowing more CPU processing power to be dedicated to packet forwarding. IPv4 Cisco Express Forwarding and distributed Cisco Express Forwarding are enabled by default. IPv6 Cisco Express Forwarding and distributed Cisco Express Forwarding are disabled by default, but automatically enabled when you configure IPv6 routing.

IPv6 Cisco Express Forwarding and distributed Cisco Express Forwarding are automatically disabled when IPv6 routing is unconfigured. IPv6 Cisco Express Forwarding and distributed Cisco Express Forwarding cannot be disabled through configuration. You can verify the IPv6 state by entering the **show ipv6 cef** privileged EXEC command.

To route IPv6 unicast packets, you must first globally configure forwarding of IPv6 unicast packets by using the **ipv6 unicast-routing** global configuration command, and you must configure an IPv6 address and IPv6 processing on an interface by using the **ipv6 address** interface configuration command.

For more information about configuring Cisco Express Forwarding and distributed Cisco Express Forwarding, see *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring Static Routing for IPv6

For more information about configuring static IPv6 routing, see the “Implementing Static Routes for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

To configure static IPv6 routing, perform this procedure:

Before you begin

You must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on at least one Layer 3 interface by configuring an IPv6 address on the interface.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 route <i>ipv6-prefix/prefix length {ipv6-address interface-id [ipv6-address]} [administrative distance]</i> Example: Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130	Configures a static IPv6 route. <ul style="list-style-type: none"> • <i>ipv6-prefix</i>—The IPv6 network that is the destination of the static route. It can also be a hostname when static host routes are configured. • <i>/prefix length</i>—The length of the IPv6 prefix. A decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value. • <i>ipv6-address</i>—The IPv6 address of the next hop that can be used to reach the specified network. The IPv6 address of the next hop need not be directly connected; recursion is done to find the IPv6 address of the directly connected next hop. The address must be in the form documented in RFC 2373, specified in hexadecimal using 16-bit values between colons. • <i>interface-id</i>—Specifies direct static routes from point-to-point and broadcast interfaces. With point-to-point interfaces, there is no need to specify the IPv6 address of the next hop. With broadcast interfaces, you should always specify the IPv6 address of the next hop, or ensure that the specified prefix is assigned to the link, specifying a link-local address as the next hop. You can optionally specify the IPv6 address of the next hop to which packets are sent.

	Command or Action	Purpose
		<p>Note You must specify an <i>interface-id</i> when using a link-local address as the next hop (the link-local next hop must also be an adjacent router).</p> <ul style="list-style-type: none"> • <i>administrative distance</i>—(Optional) An administrative distance. The range is 1 to 254; the default value is 1, which gives static routes precedence over any other type of route except connected routes. To configure a floating static route, use an administrative distance greater than that of the dynamic routing protocol.
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>Use one of the following:</p> <ul style="list-style-type: none"> • show ipv6 static [<i>ipv6-address</i> <i>ipv6-prefix/prefix length</i>] [interface <i>interface-id</i>] [detail][recursive] [detail] • show ipv6 route static [<i>updated</i>] <p>Example:</p> <pre>Switch# show ipv6 static 2001:0DB8::/32 interface gigabitethernet2/0/1</pre> <p>or</p> <pre>Switch# show ipv6 route static</pre>	<p>Verifies your entries by displaying the contents of the IPv6 routing table.</p> <ul style="list-style-type: none"> • interface <i>interface-id</i>—(Optional) Displays only those static routes with the specified interface as an egress interface. • recursive—(Optional) Displays only recursive static routes. The recursive keyword is mutually exclusive with the interface keyword, but it can be used with or without the IPv6 prefix included in the command syntax. • detail—(Optional) Displays this additional information: <ul style="list-style-type: none"> • For valid recursive routes, the output path set, and maximum resolution depth. • For invalid routes, the reason why the route is not valid.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring RIP for IPv6

For more information about configuring RIP routing for IPv6, see the “Implementing RIP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com,

To configure RIP routing for IPv6, perform this procedure:

Before you begin

Before configuring the switch to run IPv6 RIP, you must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on any Layer 3 interfaces on which IPv6 RIP is to be enabled.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 router rip name Example: Switch(config)# ipv6 router rip cisco	Configures an IPv6 RIP routing process, and enters router configuration mode for the process.
Step 4	maximum-paths number-paths Example: Switch(config-router)# maximum-paths 6	(Optional) Define the maximum number of equal-cost routes that IPv6 RIP can support. The range is from 1 to 32, and the default is 16 routes.
Step 5	exit Example: Switch(config-router)# exit	Returns to global configuration mode.
Step 6	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.

	Command or Action	Purpose
Step 7	<p>ipv6 rip <i>name</i> enable</p> <p>Example:</p> <pre>Switch(config-if)# ipv6 rip cisco enable</pre>	Enables the specified IPv6 RIP routing process on the interface.
Step 8	<p>ipv6 rip <i>name</i> default-information {only originate}</p> <p>Example:</p> <pre>Switch(config-if)# ipv6 rip cisco default-information only</pre>	<p>(Optional) Originates the IPv6 default route (::/0) into the RIP routing process updates sent from the specified interface.</p> <p>Note To avoid routing loops after the IPv6 default route (::/0) is originated from any interface, the routing process ignores all default routes received on any interface.</p> <ul style="list-style-type: none"> • only—Select to originate the default route, but suppress all other routes in the updates sent on this interface. • originate—Select to originate the default route in addition to all other routes in the updates sent on this interface.
Step 9	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	<p>Use one of the following:</p> <ul style="list-style-type: none"> • show ipv6 rip [<i>name</i>] [interface <i>interface-id</i>] [database] [next-hops] • show ipv6 rip <p>Example:</p> <pre>Switch# show ipv6 rip cisco interface gigabitethernet 2/0/1</pre> <p>or</p> <pre>Switch# show ipv6 rip</pre>	<ul style="list-style-type: none"> • Displays information about current IPv6 RIP processes. • Displays the current contents of the IPv6 routing table.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring OSPF for IPv6

For more information about configuring OSPF routing for IPv6, see the “Implementing OSPF for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

To configure OSPF routing for IPv6, perform this procedure:

Before you begin

You can customize OSPF for IPv6 for your network. However, the defaults for OSPF in IPv6 are set to meet the requirements of most customers and features.

Follow these guidelines:

- Be careful when changing the defaults for IPv6 commands. Changing the defaults might adversely affect OSPF for the IPv6 network.
- Before you enable IPv6 OSPF on an interface, you must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on Layer 3 interfaces on which you are enabling IPv6 OSPF.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 router ospf process-id Example: Switch(config)# ipv6 router ospf 21	Enables OSPF router configuration mode for the process. The process ID is the number assigned administratively when enabling the OSPF for IPv6 routing process. It is locally assigned and can be a positive integer from 1 to 65535.
Step 4	area area-id range {ipv6-prefix/prefix length} [advertise not-advertise] [cost cost] Example: Switch(config)# area .3 range 2001:0DB8::/32 not-advertise	(Optional) Consolidates and summarizes routes at an area boundary. <ul style="list-style-type: none"> • <i>area-id</i>—Identifier of the area about which routes are to be summarized. It can be specified as either a decimal value or as an IPv6 prefix. • <i>ipv6-prefix/prefix length</i>—The destination IPv6 network and a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the

	Command or Action	Purpose
		<p>address). A slash mark (/) must precede the decimal value.</p> <ul style="list-style-type: none"> • advertise—(Optional) Sets the address range status to advertise and generate a Type 3 summary link-state advertisement (LSA). • not-advertise—(Optional) Sets the address range status to DoNotAdvertise. The Type 3 summary LSA is suppressed, and component networks remain hidden from other networks. • cost cost—(Optional) Sets the metric or cost for this summary route, which is used during OSPF SPF calculation to determine the shortest paths to the destination. The value can be 0 to 16777215.
Step 5	<p>maximum paths <i>number-paths</i></p> <p>Example:</p> <pre>Switch(config)# maximum paths 16</pre>	(Optional) Defines the maximum number of equal-cost routes to the same destination that IPv6 OSPF should enter in the routing table. The range is from 1 to 32, and the default is 16 paths.
Step 6	<p>exit</p> <p>Example:</p> <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 7	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 8	<p>ipv6 ospf <i>process-id</i> area <i>area-id</i> [instance <i>instance-id</i>]</p> <p>Example:</p> <pre>Switch(config-if)# ipv6 ospf 21 area .3</pre>	<p>Enables OSPF for IPv6 on the interface.</p> <ul style="list-style-type: none"> • instance <i>instance-id</i>—(Optional) Instance identifier.
Step 9	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	<p>Use one of the following:</p> <ul style="list-style-type: none"> • show ipv6 ospf [<i>process-id</i>] [<i>area-id</i>] interface [<i>interface-id</i>] 	<ul style="list-style-type: none"> • Displays information about OSPF interfaces. • Displays general information about OSPF routing processes.

	Command or Action	Purpose
	<ul style="list-style-type: none"> • <code>show ipv6 ospf [process-id] [area-id]</code> <p>Example:</p> <pre>Switch# show ipv6 ospf 21 interface gigabitethernet2/0/1</pre> <p>OR</p> <pre>Switch# show ipv6 ospf 21</pre>	
Step 11	<p><code>copy running-config startup-config</code></p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Tuning LSA and SPF Timers for OSPFv3 Fast Convergence

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ipv6 router ospf`*process-id*
4. `timers lsa arrival` *milliseconds*
5. `timers pacing flood`*milliseconds*
6. `timers pacing lsa-group`*seconds*
7. `timers pacing retransmission`*milliseconds*
8. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<code>ipv6 router ospf</code> <i>process-id</i>	Enables OSPFv3 router configuration mode.

	Command or Action	Purpose
Step 4	<code>timers lsa arrival milliseconds</code>	Sets the minimum interval at which the software accepts the same LSA from OSPFv3 neighbors.
Step 5	<code>timers pacing flood milliseconds</code>	Configures LSA flood packet pacing.
Step 6	<code>timers pacing lsa-group seconds</code>	Changes the interval at which OSPFv3 LSAs are collected into a group and refreshed, checksummed, or aged.
Step 7	<code>timers pacing retransmission milliseconds</code>	Configures LSA retransmission packet pacing in OSPFv3.
Step 8	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring LSA and SPF Throttling for OSPFv3 Fast Convergence

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ipv6 router ospf process-id`
4. `timers throttle spf spf-start spf-hold spf-max-wait`
5. `timers throttle lsa start-interval hold-interval max-interval`
6. `timers lsa arrival milliseconds`
7. `timers pacing flood milliseconds`
8. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<code>ipv6 router ospf process-id</code>	Enables OSPFv3 router configuration mode.
Step 4	<code>timers throttle spf spf-start spf-hold spf-max-wait</code>	Turns on SPF throttling.

	Command or Action	Purpose
Step 5	<code>timers throttle lsastart-intervalhold-intervalmax-interval</code>	Sets rate-limiting values for OSPFv3 LSA generation.
Step 6	<code>timers lsa arrivalmilliseconds</code>	Sets the minimum interval at which the software accepts the same LSA from OSPFv3 neighbors.
Step 7	<code>timers pacing floodmilliseconds</code>	Configures LSA flood packet pacing.
Step 8	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring EIGRP for IPv6

Before configuring the switch to run IPv6 EIGRP, enable routing by entering the **ip routing global configuration** command, enable the forwarding of IPv6 packets by entering the **ipv6 unicast-routing global configuration** command, and enable IPv6 on any Layer 3 interfaces on which you want to enable IPv6 EIGRP.

To set an explicit router ID, use the **show ipv6 eigrp** command to see the configured router IDs, and then use the **router-id** command.

As with EIGRP IPv4, you can use EIGRPv6 to specify your EIGRP IPv6 interfaces and to select a subset of those as passive interfaces. Use the **passive-interface** command to make an interface passive, and then use the **no passive-interface** command on selected interfaces to make them active. EIGRP IPv6 does not need to be configured on a passive interface.

For more configuration procedures, see the “Implementing EIGRP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring HSRP for IPv6

Hot Standby Router Protocol (HSRP) for IPv6 provides routing redundancy for routing IPv6 traffic not dependent on the availability of any single router.

When HSRP for IPv6 is enabled on a switch, IPv6 hosts learn of available IPv6 routers through IPv6 neighbor discovery router advertisement messages. An HSRP IPv6 group has a virtual MAC address that is derived from the HSRP group number. The group has a virtual IPv6 link-local address that is, by default, derived from the HSRP virtual MAC address. Periodic messages are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active.

When configuring HSRP for IPv6, you must enable HSRP version 2 (HSRPv2) on the interface.



Note Before configuring an HSRP for IPv6 group, you must enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command and enable IPv6 on the interface on which you will configure an HSRP for IPv6 group.

Enabling HSRP Version 2

For more information about configuring HSRP for IPv6, see the “Configuring First Hop Redundancy Protocols in IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and enters the Layer 3 interface on which you want to specify the standby version.
Step 3	standby version {1 2} Example: Switch(config-if)# standby version 2	Sets the HSRP version. Enter 2 to change the HSRP version. The default is 1.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show standby Example: Switch# show standby	Verifies the configuration.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling an HSRP Group for IPv6

This task explains how to create or enable HSRP for IPv6 on a Layer 3 interface.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and enters the Layer 3 interface on which you want to enable HSRP for IPv6.
Step 3	standby [group-number] ipv6 {link-local-address autoconfig} Example: Switch(config-if)# standby 2 ipv6 auto config	Creates (or enables) the HSRP for IPv6 group. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number on the interface for which HSRP is being enabled. The range is 0 to 4095. The default is 0. If there is only one HSRP group, you do not need to enter a group number. • Enter the link-local address of the Hot Standby router interface, or enable the link-local address to be generated automatically from the link-local prefix and a modified EUI-64 format interface identifier, where the EUI-64 interface identifier is created from the relevant HSRP virtual MAC address.
Step 4	standby [group-number] preempt [delay {minimum seconds reload seconds sync seconds}] Example: Switch(config-if)# standby 2 preempt delay reload 0	Configures the router to preempt , which means that when the local router has a higher priority than the active router, it assumes control as the active router. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies. • (Optional) delay—Sets to cause the local router to postpone taking over the active role for the shown number of seconds. The range is 0 to 3600 (1 hour). The default is 0 (no delay before taking over). • (Optional) reload—Sets the preemption delay, in seconds, after a reload. The delay period applies only to the first interface-up event after the router reloads. • (Optional) sync—Sets the maximum synchronization period, in seconds, for IP redundancy clients. Use the no form of the command to restore the default values.

	Command or Action	Purpose
Step 5	standby [<i>group-number</i>] priority <i>priority</i> Example: Switch(config-if)# standby 2 priority 200	Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. Use the no form of the command to restore the default values.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show standby [<i>interface-id</i> [<i>group-number</i>]] Example: Switch# show standby gigabitethernet 1/0/1 2	Verifies the configuration.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Multi-VRF CE

The switch supports multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE) when it is running the IP services or advanced IP Services feature set. Multi-VRF CE allows a service provider to support two or more VPNs with overlapping IP addresses.



Note The switch does not use Multiprotocol Label Switching (MPLS) to support VPNs.

IPv6 multicast routing is not supported on a VRF associated interface.

Default Multi-VRF CE Configuration

Table 36: Default VRF Configuration

Feature	Default Setting
VRF	Disabled. No VRFs are defined.
Maps	No import maps, export maps, or route maps are defined.
Forwarding table	The default for an interface is the global routing table.

Configuring VRFs

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference*.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ipv6 unicast-routing Example: Switch(config)# ipv6 unicast routing	Enables IPv6 unicast routing.
Step 3	vrf definition vrf-name Example: Switch(config)# vrf definition vpn1	Names the VRF, and enters VRF configuration mode.
Step 4	address family ipv6 Example: Switch(config)# address family ipv6	Specifies the IPv6 address family and enter address family configuration mode.
Step 5	rd route-distinguisher Example: Switch(config-vrf)# rd 100:2	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y)
Step 6	route-target {export import both} <i>route-target-ext-community</i> Example: Switch(config-vrf)# route-target both 100:2	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The <i>route-target-ext-community</i> should be the same as the <i>route-distinguisher</i> entered in Step 4.
Step 7	import map route-map Example: Switch(config-vrf)# import map importmap1	(Optional) Associates a route map with the VRF.

	Command or Action	Purpose
Step 8	interface <i>interface-id</i> Example: <pre>Switch(config-vrf)# interface gigabitethernet 1/0/1</pre>	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or SVI.
Step 9	vrf forwarding <i>vrf-name</i> Example: <pre>Switch(config-if)# vrf forwarding vpn1</pre>	Associates the VRF with the Layer 3 interface.
Step 10	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 11	show vrf [brief detail interfaces] [<i>vrf-name</i>] Example: <pre>Switch# show vrf interfaces vpn1</pre>	Verifies the configuration. Displays information about the configured VRFs.
Step 12	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring VRF-Aware Services

These services are VRF-Aware:

- ARP
- Ping
- Simple Network Management Protocol (SNMP)
- Hot Standby Router Protocol (HSRP)
- Unicast Reverse Path Forwarding (uRPF)
- Syslog
- Traceroute
- FTP and TFTP



Note The switch does not support VRF-aware services for Unicast Reverse Path Forwarding (uRPF) or Network Time Protocol (NTP).

Configuring VRF-Aware Services for Neighbor Discovery

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

Procedure

	Command or Action	Purpose
Step 1	show ipv6 neighbors vrfvrf-name Example: Switch# show ipv6 neighbors vrf vpn1	Displays the ARP table in the specified VRF.

Configuring VRF-Aware Services for PING

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release .*

Procedure

	Command or Action	Purpose
Step 1	ping vrfvrf-nameipv6ipv6-address Example: Switch# ping vrf vpn1 ipv6	Displays the ARP table in the specified VRF.

Configuring VRF-Aware Services for HSRP

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	interface <i>interface-id</i> Example: Switch# interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	no switchport Example: Switch# no switchport	Removes the interface from Layer 2 configuration mode if it is a physical interface.
Step 4	vrf forwarding <i>vrf-name</i> Example: Switch# vrf forwarding vpn1	Configures VRF on the interface.
Step 5	ipv6 address <i>ipv6 address</i> Example: Switch# ipv6 address 2001::DB8:1/64	Enters the IPv6 address for the interface.
Step 6	standby 1 ipv6 <i>ipv6 address</i> Example: Switch# standby 1 ipv6 2001::DB8:1/64	Enables HSRP and configures the virtual IP address.
Step 7	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Configuring VRF-Aware Services for Traceroute

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release* .

Procedure

	Command or Action	Purpose
Step 1	traceroute vrf <i>vrf-name</i> <i>ipv6-address</i> Example: Switch# traceroute vrf vpn1 2001::DB8:1/64	Specifies the name of a VPN VRF in which to find the destination address.

Configuring VRF-Aware Services for FTP and TFTP

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ip ftp source-interface <i>interface-type interface-number</i> Example: Switch(config)# ip ftp source-interface gigabitethernet 1/0/2	Specifies the source IP address for FTP connections.
Step 3	end Example: Switch(config)#end	Returns to privileged EXEC mode.
Step 4	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 5	ip tftp source-interface <i>interface-type interface-number</i> Example: Switch(config)# ip tftp source-interface gigabitethernet 1/0/2	Specifies the source IP address for TFTP connections.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	end Example: Switch(config)#end	Returns to privileged EXEC mode.

Configuring a VPN Routing Session

Routing within the VPN can be configured with any supported routing protocol (OSPF, EIGRP, or BGP) or with static routing. The configuration shown here is for OSPF, but the process is the same for other protocols.



Note To configure an EIGRP routing process to run within a VRF instance, you must configure an autonomous-system number by entering the **autonomous-system** *autonomous-system-number* address-family configuration mode command.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router ospfv3 process-id Example: Switch(config)# router ospfv3 1	Enables OSPF routing, specifies a VPN forwarding table, and enter router configuration mode.
Step 3	router router-id Example: Switch(config)# router router-id	Specifies the OSPF router-id in IP address format for this OSPFv3 process.
Step 4	log-adjacency-changes Example: Switch(config-router)# log-adjacency-changes	(Optional) Logs changes in the adjacency state. This is the default state.
Step 5	address-family ipv6 unicast vrf vrf-name Example: Switch(config-router)# address-family ipv6 unicast vrf vpn1	Enters address family command mode for the VRF.
Step 6	area area-id normal Example: Switch(config-router)# area 2	Specifies OSPFv3 area parameters and type.
Step 7	redistribute bgp autonomous-system-number Example: Switch(config-router)# redistribute bgp 10	Redistributes routes from BGP routing process to OSPF routing process.
Step 8	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	<code>Switch(config-router)# end</code>	
Step 9	show ospfv3 vrf <i>vrf-name</i> Example: <code>Switch# show ospfv3 vrf vpn1</code>	Verifies the configuration of the OSPFv3 network.
Step 10	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring BGP PE to CE Routing Sessions

Procedure

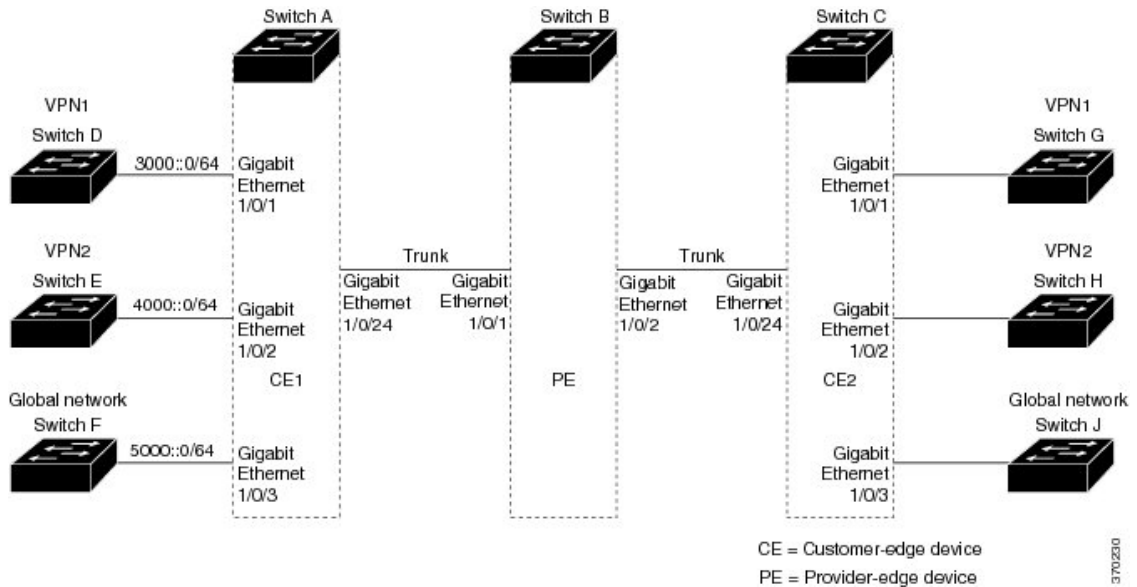
	Command or Action	Purpose
Step 1	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 2	router bgp <i>autonomous-system-number</i> Example: <code>Switch(config)# router bgp 2</code>	Configures the BGP routing process with the AS number passed to other BGP routers, and enter router configuration mode.
Step 3	bgp router id <i>router-id</i> Example: <code>Switch(config)# bgp router-id</code>	Configures a fixed 32-bit router id as the identifier of the local router running BGP.
Step 4	redistribute ospf <i>process-id</i> Example: <code>Switch(config-router)# redistribute ospf 1</code>	Sets the switch to redistribute OSPF internal routes.
Step 5	address-family ipv6 vrf <i>vrf-name</i> Example: <code>Switch(config-router)# address-family ipv6 vrf vpn1</code>	Defines BGP parameters for PE to CE routing sessions, and enter VRF address-family mode.
Step 6	network <i>ipv6 network-number</i> Example:	Specifies an IPv6 Network number to announce via BGP.

	Command or Action	Purpose
	<code>Switch(config-router)# network ipv6 255.255.255.0</code>	
Step 7	neighbor ipv6 address remote-as as-number Example: <code>Switch(config-router)# neighbor 10.1.1.2 remote-as 2</code>	Defines a BGP session between PE and CE routers.
Step 8	neighbor address activate Example: <code>Switch(config-router)# neighbor 10.2.1.1 activate</code>	Activates the advertisement of the IPv4 address family.
Step 9	end Example: <code>Switch(config-router)# end</code>	Returns to privileged EXEC mode.
Step 10	show bgp vrf vrf-name Example: <code>Switch# show ip bgp ipv4 neighbors</code>	Verifies BGP configuration on the VRF.
Step 11	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Multi-VRF CE Configuration Example

OSPF is the protocol used in VPN1, VPN2, and the global network. BGP is used in the CE to PE connections. The examples following the illustration show how to configure a switch as CE Switch A, and the VRF configuration for customer switches D and E. Commands for configuring CE Switch C and the other customer switches are not included but would be similar.

Figure 14: Multi-VRF CE Configuration Example



On Switch A, enable routing and configure VRF.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# ipv6 unicast-routing
Switch(config)# vrf definition v11
Switch(config-vrf)# rd 11:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf)# exit
Switch(config-vrf)# vrf definition v12
Switch(config-vrf)# rd 12:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf-af)# end
```

Configure the physical interfaces on Switch A. Gigabit Ethernet interface 1/0/24 is a trunk connection to the PE. Gigabit Ethernet ports 1/0/1 and 1/0/2 connect to VPNs.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface GigabitEthernet 1/0/1
Switch(config-if)# switchport access vlan 208
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface gigabitEthernet 1/0/2
Switch(config-if)# switchport access vlan 118
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet 1/0/24
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# exit
```

Configure the VLANs used on Switch A. VLAN 10 is used by VRF 11 between the CE and the PE. VLAN 20 is used by VRF 12 between the CE and the PE. VLANs 118 and 208 are used for the VPNs that include Switch E and Switch D, respectively:

```
Switch(config)# interface vlan10
Switch(config-if)# vrf forwarding v11
Switch(config-if)# ipv6 address 1000::1/64
Switch(config-if)# exit
```

```
Switch(config)# interface vlan20
Switch(config-if)# vrf forwarding v12
Switch(config-if)# ipv6 address 2000::1/64
Switch(config-if)# exit
```

```
Switch(config)# interface vlan208
Switch(config-if)# vrf forwarding v11
Switch(config-if)# ipv6 address 3000::1/64
Switch(config-if)# exit
```

```
Switch(config)# interface vlan118
Switch(config-if)# vrf forwarding v12
Switch(config-if)# ipv6 address 4000::1/64
Switch(config-if)# exit
```

Configure OSPFv3 routing on VPN1 and VPN2.

```
Switch(config)# router ospfv3 1
Switch(config-router)# router-id 10.1.1.10
Switch(config-router)# address-family ipv6 unicast vrf v11
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute bgp 800
Switch(config-router-af)# exit
Switch(config)# router ospfv3 2
Switch(config-router)# router-id 2.2.2.2
Switch(config-router)# address-family ipv6 unicast vrf v12
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute bgp 800
Switch(config-router-af)# exit
Switch(config-router)# exit
Switch(config)# exit
```

Configure BGP for CE to PE routing.

```
Switch(config)# router bgp 800
Switch(config-router)# bgp router-id 8.8.8.8
Switch(config-router)# address-family ipv6 vrf v11
Switch(config-router-af)# redistribute ospf 1
Switch(config-router-af)# neighbor 1000::2 remote-as 100
Switch(config-router-af)# neighbor 1000::2 activate
Switch(config-router-af)# network 3000::/64
Switch(config-router-af)# exit
```

```
Switch(config)# address-family ipv6 vrf v12
Switch(config-router-af)# redistribute ospf 2
Switch(config-router-af)# neighbor 2000::2 remote-as 100
Switch(config-router-af)# neighbor 2000::2 activate
Switch(config-router-af)# network 4000::/64
```

Switch D belongs to VPN 1. Configure the connection to Switch A by using these commands.

```

Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# ipv6 unicast-routing
Switch(config)# interface GigabitEthernet 5/0/16
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 3000::2/64
Switch(config-if)# exit

Switch(config-router)# router ospfv3 101
Switch(config-router)# address-family ipv6
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute connected
Switch(config-router-af)# exit
Switch(config-router)# exit

```

Switch E belongs to VPN 2. Configure the connection to Switch A by using these commands.

```

Switch(config)# ipv6 unicast-routing
Switch(config)# interface GigabitEthernet 3/0/13
Switch(config-if)# switchport access vlan 20
Switch(config-if)# exit
Switch(config)# interface vlan 20
Switch(config-if)# ipv6 address 4000::2/64

```

```

Switch(config)# router ospfv3 101
Switch(config-router)# address-family ipv6
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute connected
Switch(config-router-af)# end

```

When used on switch B (the PE router), these commands configure only the connections to the CE device, Switch A.

```

Switch(config)# vrf definition v1
Switch(config-vrf)# rd 1:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf-af)# exit
Switch(config-vrf)# exit

Switch(config)# vrf definition v2
Switch(config-vrf)# rd 2:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf-af)# exit
Switch(config-vrf)# exit

Switch(config-if)# interface g 1/0/2
Switch(config-if)# vrf forwarding v1
Switch(config-if)# ipv6 address 1000::2/64
Switch(config-if)# exit
Switch(config)# interface g 1/0/4
Switch(config-if)# vrf forwarding v2
Switch(config-if)# ipv6 address 2000::2/64

Switch(config-if)# interface gigabitEthernet 1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk

Switch(config)# router bgp 100
Switch(config-router)# address-family ipv6 vrf v1
Switch(config-router-af)# neighbor 1000::1 remote-as 100

```

```
Switch(config-router-af)# neighbor 1000::1 activate
Switch(config-router-af)# network 3000::/64
Switch(config-router-af)# exit
Switch(config-router)# address-family ipv6 vrf v2
Switch(config-router-af)# neighbor 2000::1 remote-as 100
Switch(config-router-af)# neighbor 2000::1 activate
Switch(config-router-af)# network 4000::/64
```

Displaying Multi-VRF CE Status

Table 37: Commands for Displaying Multi-VRF CE Information

Command	Purpose
show ipv6 protocols vrf <i>vrf-name</i>	Displays routing protocol information associated with a VRF.
show ipv6 route vrf <i>vrf-name</i> [connected] [<i>protocol</i> [<i>as-number</i>]] [list] [mobile] [odr] [profile] [static] [summary] [supernets-only]	Displays IP routing table information associated with a VRF.
show ipv6 vrf [brief detail interfaces] [<i>vrf-name</i>]	Displays information about the defined VRF instances.

Displaying IPv6

For complete syntax and usage information on these commands, see the Cisco IOS command reference publications.

Table 38: Command for Monitoring IPv6

Command	Purpose
show ipv6 access-list	Displays a summary of access lists.
show ipv6 cef	Displays Cisco Express Forwarding for IPv6.
show ipv6 interface <i>interface-id</i>	Displays IPv6 interface status and configuration.
show ipv6 mtu	Displays IPv6 MTU per destination cache.
show ipv6 neighbors	Displays IPv6 neighbor cache entries.
show ipv6 prefix-list	Displays a list of IPv6 prefix lists.
show ipv6 protocols	Displays a list of IPv6 routing protocols on the switch.
show ipv6 rip	Displays IPv6 RIP routing protocol status.
show ipv6 route	Displays IPv6 route table entries.
show ipv6 static	Displays IPv6 static routes.
show ipv6 traffic	Displays IPv6 traffic statistics.

Configuring DHCP for IPv6 Address Assignment

This section describes only the DHCPv6 address assignment. For more information about configuring the DHCPv6 client, server, or relay agent functions, see the “Implementing DHCP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Default DHCPv6 Address Assignment Configuration

By default, no DHCPv6 features are configured on the switch.

DHCPv6 Address Assignment Configuration Guidelines

When configuring DHCPv6 address assignment, consider these guidelines:

- In the procedures, the specified interface must be one of these Layer 3 interfaces:
 - DHCPv6 IPv6 routing must be enabled on a Layer 3 interface.
 - SVI: a VLAN interface created by using the **interface vlan** *vlan_id* command.
 - EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* command.
- The switch can act as a DHCPv6 client, server, or relay agent. The DHCPv6 client, server, and relay function are mutually exclusive on an interface.

Enabling DHCPv6 Server Function (CLI)

Use the **no** form of the DHCP pool configuration mode commands to change the DHCPv6 pool characteristics. To disable the DHCPv6 server function on an interface, use the **no ipv6 dhcp server** interface configuration command.

To enable the DHCPv6 server function on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ipv6 dhcp pool <i>poolname</i> Example: Switch(config)# ipv6 dhcp pool 7	Enters DHCP pool configuration mode, and define the name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0).
Step 4	address prefix <i>IPv6-prefix</i> { lifetime } { <i>t1 t1</i> infinite } Example: Switch(config-dhcpv6)# address prefix 2001:1002::0/64 lifetime 3600	(Optional) Specifies an address prefix for address assignment. This address must be in hexadecimal, using 16-bit values between colons. lifetime <i>t1 t1</i> —Specifies a time interval (in seconds) that an IPv6 address prefix remains in the valid state. The range is 5 to 4294967295 seconds. Specify infinite for no time interval.
Step 5	link-address <i>IPv6-prefix</i> Example: Switch(config-dhcpv6)# link-address 2001:1002::0/64	(Optional) Specifies a link-address IPv6 prefix. When an address on the incoming interface or a link-address in the packet matches the specified IPv6 prefix, the server uses the configuration information pool. This address must be in hexadecimal, using 16-bit values between colons.
Step 6	vendor-specific <i>vendor-id</i> Example: Switch(config-dhcpv6)# vendor-specific 9	(Optional) Enters vendor-specific configuration mode and specifies a vendor-specific identification number. This number is the vendor IANA Private Enterprise Number. The range is 1 to 4294967295.
Step 7	suboption <i>number</i> { address <i>IPv6-address</i> ascii <i>ASCII-string</i> hex <i>hex-string</i> } Example: Switch(config-dhcpv6-vs)# suboption 1 address 1000:235D::	(Optional) Enters a vendor-specific suboption number. The range is 1 to 65535. Enter an IPv6 address, ASCII text, or a hex string as defined by the suboption parameters.
Step 8	exit Example: Switch(config-dhcpv6-vs)# exit	Returns to DHCP pool configuration mode.
Step 9	exit Example: Switch(config-dhcpv6)# exit	Returns to global configuration mode.

	Command or Action	Purpose
Step 10	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode, and specifies the interface to configure.
Step 11	ipv6 dhcp server [<i>poolname</i> automatic] [rapid-commit] [preference <i>value</i>] [allow-hint] Example: <pre>Switch(config-if)# ipv6 dhcp server automatic</pre>	Enables DHCPv6 server function on an interface. <ul style="list-style-type: none"> • poolname—(Optional) User-defined name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0). • automatic—(Optional) Enables the system to automatically determine which pool to use when allocating addresses for a client. • rapid-commit—(Optional) Allows two-message exchange method. • preference value—(Optional) Configures the preference value carried in the preference option in the advertise message sent by the server. The range is from 0 to 255. The preference value default is 0. • allow-hint—(Optional) Specifies whether the server should consider client suggestions in the SOLICIT message. By default, the server ignores client hints.
Step 12	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 13	Do one of the following: <ul style="list-style-type: none"> • show ipv6 dhcp pool • show ipv6 dhcp interface Example: <pre>Switch# show ipv6 dhcp pool</pre> or <pre>Switch# show ipv6 dhcp interface</pre>	<ul style="list-style-type: none"> • Verifies DHCPv6 pool configuration. • Verifies that the DHCPv6 server function is enabled on an interface.
Step 14	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling DHCPv6 Client Function

To enable the DHCPv6 client on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ipv6 address dhcp [rapid-commit] Example: Switch(config-if)# ipv6 address dhcp rapid-commit	Enables the interface to acquire an IPv6 address from the DHCPv6 server. rapid-commit —(Optional) Allow two-message exchange method for address assignment.
Step 5	ipv6 dhcp client request [vendor-specific] Example: Switch(config-if)# ipv6 dhcp client request vendor-specific	(Optional) Enables the interface to request the vendor-specific option.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show ipv6 dhcp interface Example: Switch# show ipv6 dhcp interface	Verifies that the DHCPv6 client is enabled on an interface.

Configuration Examples for IPv6 Unicast Routing

Configuring IPv6 Addressing and Enabling IPv6 Routing: Example

This example shows how to enable IPv6 with both a link-local address and a global address based on the IPv6 prefix 2001:0DB8:c18:1::/64. The EUI-64 interface ID is used in the low-order 64 bits of both addresses. Output from the **show ipv6 interface EXEC** command is included to show how the interface ID (20B:46FF:FE2F:D940) is appended to the link-local prefix FE80::/64 of the interface.

```
Switch(config)# ipv6 unicast-routing
Switch(config)# interface gigabitethernet0/11

Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
Switch# show ipv6 interface gigabitethernet0/11
GigabitEthernet0/11 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/64 [EUI]
  Joined group address(es):
    FE02::1
    FE02::2
    FE02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
```

Configuring Default Router Preference: Example

This example shows how to configure a DRP of *high* for the router on an interface.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ipv6 nd router-preference high
Switch(config-if)# end
```

Enabling an HSRP Group for IPv6: Example

This example shows how to activate HSRP for IPv6 for group 1 on a port. The IP address used by the hot standby group is learned by using HSRP for IPv6.



Note This procedure is the minimum number of steps required to enable HSRP for IPv6. Other configurations are optional.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ipv6 autoconfig
Switch(config-if)# end
Switch# show standby
```

Enabling DHCPv6 Server Function: Example

This example shows how to configure a pool called *engineering* with an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool engineering
Switch(config-dhcpv6)# address prefix 2001:1000::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *testgroup* with three link-addresses and an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool testgroup
Switch(config-dhcpv6)# link-address 2001:1001::0/64
Switch(config-dhcpv6)# link-address 2001:1002::0/64
Switch(config-dhcpv6)# link-address 2001:2000::0/48
Switch(config-dhcpv6)# address prefix 2001:1003::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *350* with vendor-specific options:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool 350
Switch(config-dhcpv6)# address prefix 2001:1005::0/48
Switch(config-dhcpv6)# vendor-specific 9
Switch(config-dhcpv6-vs)# suboption 1 address 1000:235D::1
Switch(config-dhcpv6-vs)# suboption 2 ascii "IP-Phone"
Switch(config-dhcpv6-vs)# end
```

Enabling DHCPv6 Client Function: Example

This example shows how to acquire an IPv6 address and to enable the rapid-commit option:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ipv6 address dhcp rapid-commit
```

Configuring IPv6 ICMP Rate Limiting: Example

This example shows how to configure an IPv6 ICMP error message interval of 50 milliseconds and a bucket size of 20 tokens.

```
Switch(config)#ipv6 icmp error-interval 50 20
```

Configuring Static Routing for IPv6: Example

This example shows how to configure a floating static route to an interface with an administrative distance of 130:

```
Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet 1/0/1 130
```

Configuring RIP for IPv6: Example

This example shows how to enable the RIP routing process *cisco* with a maximum of eight equal-cost routes and to enable it on an interface:

```
Switch(config)# ipv6 router rip cisco
Switch(config-router)# maximum-paths 8
Switch(config)# exit
Switch(config)# interface gigabitethernet2/0/11
Switch(config-if)# ipv6 rip cisco enable
```

Displaying IPv6: Example

This is an example of the output from the **show ipv6 interface** privileged EXEC command:

```
Switch# show ipv6 interface
Vlan1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
<output truncated>
```



CHAPTER 22

Implementing IPv6 Multicast

- [Finding Feature Information, on page 357](#)
- [Information About Implementing IPv6 Multicast Routing, on page 357](#)
- [Implementing IPv6 Multicast, on page 367](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Information About Implementing IPv6 Multicast Routing

This chapter describes how to implement IPv6 multicast routing on the switch.

Traditional IP communication allows a host to send packets to a single host (unicast transmission) or to all hosts (broadcast transmission). IPv6 multicast provides a third scheme, allowing a host to send a single data stream to a subset of all hosts (group transmission) simultaneously.



Note IPv6 Multicast Routing is supported only on Cisco Catalyst 3560-CX switches.

IPv6 Multicast Overview

An IPv6 multicast group is an arbitrary group of receivers that want to receive a particular data stream. This group has no physical or geographical boundaries--receivers can be located anywhere on the Internet or in any private network. Receivers that are interested in receiving data flowing to a particular group must join the group by signaling their local switch. This signaling is achieved with the MLD protocol.

Switches use the MLD protocol to learn whether members of a group are present on their directly attached subnets. Hosts join multicast groups by sending MLD report messages. The network then delivers data to a

potentially unlimited number of receivers, using only one copy of the multicast data on each subnet. IPv6 hosts that wish to receive the traffic are known as group members.

Packets delivered to group members are identified by a single multicast group address. Multicast packets are delivered to a group using best-effort reliability, just like IPv6 unicast packets.

The multicast environment consists of senders and receivers. Any host, regardless of whether it is a member of a group, can send to a group. However, only members of a group can listen to and receive the message.

A multicast address is chosen for the receivers in a multicast group. Senders use that address as the destination address of a datagram to reach all members of the group.

Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time.

How active a multicast group is, its duration, and its membership can vary from group to group and from time to time. A group that has members may have no activity.

IPv6 Multicast Routing Implementation

The Cisco IOS software supports the following protocols to implement IPv6 multicast routing:

- MLD is used by IPv6 switches to discover multicast listeners (nodes that want to receive multicast packets destined for specific multicast addresses) on directly attached links. There are two versions of MLD: MLD version 1 is based on version 2 of the Internet Group Management Protocol (IGMP) for IPv4, and MLD version 2 is based on version 3 of the IGMP for IPv4. IPv6 multicast for Cisco IOS software uses both MLD version 2 and MLD version 1. MLD version 2 is fully backward-compatible with MLD version 1 (described in RFC 2710). Hosts that support only MLD version 1 will interoperate with a switch running MLD version 2. Mixed LANs with both MLD version 1 and MLD version 2 hosts are likewise supported.
- PIM-SM is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs.
- PIM in Source Specific Multicast (PIM-SSM) is similar to PIM-SM with the additional ability to report interest in receiving packets from specific source addresses (or from all but the specific source addresses) to an IP multicast address.

MLD Access Group

The MLD access group provides receiver access control in Cisco IOS IPv6 multicast switches. This feature limits the list of groups a receiver can join, and it allows or denies sources used to join SSM channels.

Explicit Tracking of Receivers

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network. This feature also enables the fast leave mechanism to be used with MLD version 2 host reports.

IPv6 Multicast User Authentication and Profile Support

IPv6 multicast by design allows any host in the network to become a receiver or a source for a multicast group. Therefore, multicast access control is needed to control multicast traffic in the network. Access control functionality consists mainly of source access control and accounting, receiver access control and accounting, and provisioning of this access control mechanism.

Multicast access control provides an interface between multicast and authentication, authorization, and accounting (AAA) for provisioning, authorizing, and accounting at the last-hop switch, receiver access control functions in multicast, and group or channel disabling capability in multicast.

When you deploy a new multicast service environment, it is necessary to add user authentication and provide a user profile download on a per-interface basis. The use of AAA and IPv6 multicast supports user authentication and downloading of the user profile in a multicast environment.

The event that triggers the download of a multicast access-control profile from the RADIUS server to the access switch is arrival of an MLD join on the access switch. When this event occurs, a user can cause the authorization cache to time out and request download periodically or use an appropriate multicast clear command to trigger a new download in case of profile changes.

Accounting occurs via RADIUS accounting. Start and stop accounting records are sent to the RADIUS server from the access switch. In order for you to track resource consumption on a per-stream basis, these accounting records provide information about the multicast source and group. The start record is sent when the last-hop switch receives a new MLD report, and the stop record is sent upon MLD leave or if the group or channel is deleted for any reason.

IPv6 MLD Proxy

The MLD proxy feature provides a mechanism for a switch to generate MLD membership reports for all (*, G)/(S, G) entries or a user-defined subset of these entries on the switch's upstream interface. The MLD proxy feature enables a device to learn proxy group membership information, and forward multicast packets based upon that information.

If a switch is acting as RP for mroute proxy entries, MLD membership reports for these entries can be generated on user specified proxy interface.

Protocol Independent Multicast

Protocol Independent Multicast (PIM) is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs. PIM works independently of the unicast routing protocol to perform send or receive multicast route updates like other protocols. Regardless of which unicast routing protocols are being used in the LAN to populate the unicast routing table, Cisco IOS PIM uses the existing unicast table content to perform the Reverse Path Forwarding (RPF) check instead of building and maintaining its own separate routing table.

You can configure IPv6 multicast to use either PIM-SM or PIM-SSM operation, or you can use both PIM-SM and PIM-SSM together in your network.

PIM-Sparse Mode

IPv6 multicast provides support for intradomain multicast routing using PIM-SM. PIM-SM uses unicast routing to provide reverse-path information for multicast tree building, but it is not dependent on any particular unicast routing protocol.

PIM-SM is used in a multicast network when relatively few switches are involved in each multicast and these switches do not forward multicast packets for a group, unless there is an explicit request for the traffic. PIM-SM distributes information about active sources by forwarding data packets on the shared tree. PIM-SM initially uses shared trees, which requires the use of an RP.

Requests are accomplished via PIM joins, which are sent hop by hop toward the root node of the tree. The root node of a tree in PIM-SM is the RP in the case of a shared tree or the first-hop switch that is directly

connected to the multicast source in the case of a shortest path tree (SPT). The RP keeps track of multicast groups and the hosts that send multicast packets are registered with the RP by that host's first-hop switch.

As a PIM join travels up the tree, switches along the path set up multicast forwarding state so that the requested multicast traffic will be forwarded back down the tree. When multicast traffic is no longer needed, a switch sends a PIM prune up the tree toward the root node to prune (or remove) the unnecessary traffic. As this PIM prune travels hop by hop up the tree, each switch updates its forwarding state appropriately. Ultimately, the forwarding state associated with a multicast group or source is removed.

A multicast data sender sends data destined for a multicast group. The designated switch (DR) of the sender takes those data packets, unicast-encapsulates them, and sends them directly to the RP. The RP receives these encapsulated data packets, de-encapsulates them, and forwards them onto the shared tree. The packets then follow the (*, G) multicast tree state in the switches on the RP tree, being replicated wherever the RP tree branches, and eventually reaching all the receivers for that multicast group. The process of encapsulating data packets to the RP is called registering, and the encapsulation packets are called PIM register packets.

Designated Switch

Cisco switches use PIM-SM to forward multicast traffic and follow an election process to select a designated switch when there is more than one switch on a LAN segment.

The designated switch is responsible for sending PIM register and PIM join and prune messages toward the RP to inform it about active sources and host group membership.

If there are multiple PIM-SM switches on a LAN, a designated switch must be elected to avoid duplicating multicast traffic for connected hosts. The PIM switch with the highest IPv6 address becomes the DR for the LAN unless you choose to force the DR election by use of the `ipv6 pim dr-priority` command. This command allows you to specify the DR priority of each switch on the LAN segment (default priority = 1) so that the switch with the highest priority will be elected as the DR. If all switches on the LAN segment have the same priority, then the highest IPv6 address is again used as the tiebreaker.

If the DR should fail, the PIM-SM provides a way to detect the failure of Switch A and elect a failover DR. If the DR (Switch A) became inoperable, Switch B would detect this situation when its neighbor adjacency with Switch A timed out. Because Switch B has been hearing MLD membership reports from Host A, it already has MLD state for Group A on this interface and would immediately send a join to the RP when it became the new DR. This step reestablishes traffic flow down a new branch of the shared tree via Switch B. Additionally, if Host A were sourcing traffic, Switch B would initiate a new register process immediately after receiving the next multicast packet from Host A. This action would trigger the RP to join the SPT to Host A via a new branch through Switch B.



Note

- Two PIM switches are neighbors if there is a direct connection between them. To display your PIM neighbors, use the `show ipv6 pim neighbor` privileged EXEC command.
- The DR election process is required only on multiaccess LANs.

Rendezvous Point

IPv6 PIM provides embedded RP support. Embedded RP support allows the switch to learn RP information using the multicast group destination address instead of the statically configured RP. For switches that are the RP, the switch must be statically configured as the RP.

The switch searches for embedded RP group addresses in MLD reports or PIM messages and data packets. On finding such an address, the switch learns the RP for the group from the address itself. It then uses this

learned RP for all protocol activity for the group. For switches that are the RP, the switch is advertised as an embedded RP must be configured as the RP.

To select a static RP over an embedded RP, the specific embedded RP group range or mask must be configured in the access list of the static RP. When PIM is configured in sparse mode, you must also choose one or more switches to operate as an RP. An RP is a single common root placed at a chosen point of a shared distribution tree and is configured statically in each box.

PIM DRs forward data from directly connected multicast sources to the RP for distribution down the shared tree. Data is forwarded to the RP in one of two ways:

- Data is encapsulated in register packets and unicast directly to the RP by the first-hop switch operating as the DR.
- If the RP has itself joined the source tree, it is multicast-forwarded per the RPF forwarding algorithm described in the PIM-Sparse Mode section.

The RP address is used by first-hop switches to send PIM register messages on behalf of a host sending a packet to the group. The RP address is also used by last-hop switches to send PIM join and prune messages to the RP to inform it about group membership. You must configure the RP address on all switches (including the RP switch).

A PIM switch can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain for a certain group. The conditions specified by the access list determine for which groups the switch is an RP.

IPv6 multicast supports the PIM accept register feature, which is the ability to perform PIM-SM register message filtering at the RP. The user can match an access list or compare the AS path for the registered source with the AS path specified in a route map.

PIMv6 Anycast RP Solution Overview

The anycast RP solution in IPv6 PIM allows an IPv6 network to support anycast services for the PIM-SM RP. It allows anycast RP to be used inside a domain that runs PIM only. This feature is useful when interdomain connection is not required. Anycast RP can be used in IPv4 as well as IPv6, but it does not depend on the Multicast Source Discovery Protocol (MSDP), which runs only on IPv4.

Anycast RP is a mechanism that ISP-based backbones use to get fast convergence when a PIM RP device fails. To allow receivers and sources to rendezvous to the closest RP, the packets from a source need to get to all RPs to find joined receivers.

A unicast IP address is chosen as the RP address. This address is either statically configured or distributed using a dynamic protocol to all PIM devices throughout the domain. A set of devices in the domain is chosen to act as RPs for this RP address; these devices are called the anycast RP set. Each device in the anycast RP set is configured with a loopback interface using the RP address. Each device in the anycast RP set also needs a separate physical IP address to be used for communication between the RPs.

The RP address, or a prefix that covers the RP address, is injected into the unicast routing system inside of the domain. Each device in the anycast RP set is configured with the addresses of all other devices in the anycast RP set, and this configuration must be consistent in all RPs in the set.

IPv6 BSR: Configure RP Mapping

PIM switches in a domain must be able to map each multicast group to the correct RP address. The BSR protocol for PIM-SM provides a dynamic, adaptive mechanism to distribute group-to-RP mapping information rapidly throughout a domain. With the IPv6 BSR feature, if an RP becomes unreachable, it will be detected

and the mapping tables will be modified so that the unreachable RP is no longer used, and the new tables will be rapidly distributed throughout the domain.

Every PIM-SM multicast group needs to be associated with the IP or IPv6 address of an RP. When a new multicast sender starts sending, its local DR will encapsulate these data packets in a PIM register message and send them to the RP for that multicast group. When a new multicast receiver joins, its local DR will send a PIM join message to the RP for that multicast group. When any PIM switch sends a (*, G) join message, the PIM switch needs to know which is the next switch toward the RP so that G (Group) can send a message to that switch. Also, when a PIM switch is forwarding data packets using (*, G) state, the PIM switch needs to know which is the correct incoming interface for packets destined for G, because it needs to reject any packets that arrive on other interfaces.

A small set of switches from a domain are configured as candidate bootstrap switches (C-BSRs) and a single BSR is selected for that domain. A set of switches within a domain are also configured as candidate RPs (C-RPs); typically, these switches are the same switches that are configured as C-BSRs. Candidate RPs periodically unicast candidate-RP-advertisement (C-RP-Adv) messages to the BSR of that domain, advertising their willingness to be an RP. A C-RP-Adv message includes the address of the advertising C-RP, and an optional list of group addresses and mask length fields, indicating the group prefixes for which the candidacy is advertised. The BSR then includes a set of these C-RPs, along with their corresponding group prefixes, in bootstrap messages (BSMs) it periodically originates. BSMs are distributed hop-by-hop throughout the domain.

Bidirectional BSR support allows bidirectional RPs to be advertised in C-RP messages and bidirectional ranges in the BSM. All switches in a system must be able to use the bidirectional range in the BSM; otherwise, the bidirectional RP feature will not function.

PIM-Source Specific Multicast

PIM-SSM is the routing protocol that supports the implementation of SSM and is derived from PIM-SM. However, unlike PIM-SM where data from all multicast sources are sent when there is a PIM join, the SSM feature forwards datagram traffic to receivers from only those multicast sources that the receivers have explicitly joined, thus optimizing bandwidth utilization and denying unwanted Internet broadcast traffic. Further, instead of the use of RP and shared trees, SSM uses information found on source addresses for a multicast group. This information is provided by receivers through the source addresses relayed to the last-hop switches by MLD membership reports, resulting in shortest-path trees directly to the sources.

In SSM, delivery of datagrams is based on (S, G) channels. Traffic for one (S, G) channel consists of datagrams with an IPv6 unicast source address S and the multicast group address G as the IPv6 destination address. Systems will receive this traffic by becoming members of the (S, G) channel. Signaling is not required, but receivers must subscribe or unsubscribe to (S, G) channels to receive or not receive traffic from specific sources.

MLD version 2 is required for SSM to operate. MLD allows the host to provide source information. Before SSM can run with MLD, SSM must be supported in the Cisco IOS IPv6 switch, the host where the application is running, and the application itself.

SSM Mapping for IPv6

SSM mapping for IPv6 supports both static and dynamic Domain Name System (DNS) mapping for MLD version 1 receivers. This feature allows deployment of IPv6 SSM with hosts that are incapable of providing MLD version 2 support in their TCP/IP host stack and their IP multicast receiving application.

SSM mapping allows the switch to look up the source of a multicast MLD version 1 report either in the running configuration of the switch or from a DNS server. The switch can then initiate an (S, G) join toward the source.

PIM Shared Tree and Source Tree (Shortest-Path Tree)

By default, members of a group receive data from senders to the group across a single data distribution tree rooted at the RP. This type of distribution tree is called shared tree or rendezvous point tree (RPT), as illustrated in the figure below. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

If the data threshold warrants, leaf switches on the shared tree may initiate a switch to the data distribution tree rooted at the source. This type of distribution tree is called a shortest path tree or source tree. By default, the Cisco IOS software switches to a source tree upon receiving the first data packet from a source.

The following process details the move from shared tree to source tree:

1. Receiver joins a group; leaf Switch C sends a join message toward the RP.
2. RP puts the link to Switch C in its outgoing interface list.
3. Source sends the data; Switch A encapsulates the data in the register and sends it to the RP.
4. RP forwards the data down the shared tree to Switch C and sends a join message toward the source. At this point, data may arrive twice at Switch C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, the RP sends a register-stop message to Switch A.
6. By default, receipt of the first data packet prompts Switch C to send a join message toward the source.
7. When Switch C receives data on (S, G), it sends a prune message for the source up the shared tree.
8. RP deletes the link to Switch C from the outgoing interface of (S, G).
9. RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM switch along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated switch that is directly connected to a source and are received by the RP for the group.

Reverse Path Forwarding

Reverse-path forwarding is used for forwarding multicast datagrams. It functions as follows:

- If a switch receives a datagram on an interface it uses to send unicast packets to the source, the packet has arrived on the RPF interface.
- If the packet arrives on the RPF interface, a switch forwards the packet out the interfaces present in the outgoing interface list of a multicast routing table entry.
- If the packet does not arrive on the RPF interface, the packet is silently discarded to prevent loops.

PIM uses both source trees and RP-rooted shared trees to forward datagrams; the RPF check is performed differently for each, as follows:

- If a PIM switch has source-tree state (that is, an (S, G) entry is present in the multicast routing table), the switch performs the RPF check against the IPv6 address of the source of the multicast packet.
- If a PIM switch has shared-tree state (and no explicit source-tree state), it performs the RPF check on the RP's address (which is known when members join the group).

Sparse-mode PIM uses the RPF lookup function to determine where it needs to send joins and prunes. (S, G) joins (which are source-tree states) are sent toward the source. (*, G) joins (which are shared-tree states) are sent toward the RP.

Routable Address Hello Option

When an IPv6 interior gateway protocol is used to build the unicast routing table, the procedure to detect the upstream switch address assumes the address of a PIM neighbor is always same as the address of the next-hop switch, as long as they refer to the same switch. However, it may not be the case when a switch has multiple addresses on a link.

Two typical situations can lead to this situation for IPv6. The first situation can occur when the unicast routing table is not built by an IPv6 interior gateway protocol such as multicast BGP. The second situation occurs when the address of an RP shares a subnet prefix with downstream switches (note that the RP switch address has to be domain-wide and therefore cannot be a link-local address).

The routable address hello option allows the PIM protocol to avoid such situations by adding a PIM hello message option that includes all the addresses on the interface on which the PIM hello message is advertised. When a PIM switch finds an upstream switch for some address, the result of RPF calculation is compared with the addresses in this option, in addition to the PIM neighbor's address itself. Because this option includes all the possible addresses of a PIM switch on that link, it always includes the RPF calculation result if it refers to the PIM switch supporting this option.

Because of size restrictions on PIM messages and the requirement that a routable address hello option fits within a single PIM hello message, a limit of 16 addresses can be configured on the interface.

Bidirectional PIM

Bidirectional PIM allows multicast switches to keep reduced state information, as compared with unidirectional shared trees in PIM-SM. Bidirectional shared trees convey data from sources to the rendezvous point address (RPA) and distribute them from the RPA to the receivers. Unlike PIM-SM, bidirectional PIM does not switch over to the source tree, and there is no register encapsulation of data from the source to the RP.

A single designated forwarder (DF) exists for each RPA on every link within a bidirectional PIM domain (including multiaccess and point-to-point links). The only exception is the RPL on which no DF exists. The DF is the switch on the link with the best route to the RPA, which is determined by comparing MRIB-provided metrics. A DF for a given RPA forwards downstream traffic onto its link and forwards upstream traffic from its link toward the rendezvous point link (RPL). The DF performs this function for all bidirectional groups that map to the RPA. The DF on a link is also responsible for processing Join messages from downstream switches on the link as well as ensuring that packets are forwarded to local receivers discovered through a local membership mechanism such as MLD.

Bidirectional PIM offers advantages when there are many moderate or low-rate sources. However, the bidirectional shared trees may have worse delay characteristics than do the source trees built in PIM-SM (depending on the topology).

Only static configuration of bidirectional RPs is supported in IPv6.

Static Mroutes

IPv6 static mroutes behave much in the same way as IPv4 static mroutes used to influence the RPF check. IPv6 static mroutes share the same database as IPv6 static routes and are implemented by extending static route support for RPF checks. Static mroutes support equal-cost multipath mroutes, and they also support unicast-only static routes.

MRIB

The Multicast Routing Information Base (MRIB) is a protocol-independent repository of multicast routing entries instantiated by multicast routing protocols (routing clients). Its main function is to provide independence between routing protocols and the Multicast Forwarding Information Base (MFIB). It also acts as a coordination and communication point among its clients.

Routing clients use the services provided by the MRIB to instantiate routing entries and retrieve changes made to routing entries by other clients. Besides routing clients, MRIB also has forwarding clients (MFIB instances) and special clients such as MLD. MFIB retrieves its forwarding entries from MRIB and notifies the MRIB of any events related to packet reception. These notifications can either be explicitly requested by routing clients or spontaneously generated by the MFIB.

Another important function of the MRIB is to allow for the coordination of multiple routing clients in establishing multicast connectivity within the same multicast session. MRIB also allows for the coordination between MLD and routing protocols.

MFIB

The MFIB is a platform-independent and routing-protocol-independent library for IPv6 software. Its main purpose is to provide a Cisco IOS platform with an interface with which to read the IPv6 multicast forwarding table and notifications when the forwarding table changes. The information provided by the MFIB has clearly defined forwarding semantics and is designed to make it easy for the platform to translate to its specific hardware or software forwarding mechanisms.

When routing or topology changes occur in the network, the IPv6 routing table is updated, and those changes are reflected in the MFIB. The MFIB maintains next-hop address information based on the information in the IPv6 routing table. Because there is a one-to-one correlation between MFIB entries and routing table entries, the MFIB contains all known routes and eliminates the need for route cache maintenance that is associated with switching paths such as fast switching and optimum switching.

IPv6 Multicast VRF Lite

The IPv6 Multicast VRF Lite feature provides IPv6 multicast support for multiple virtual routing/forwarding contexts (VRFs). The scope of these VRFs is limited to the switch in which the VRFs are defined.

This feature provides separation between routing and forwarding, providing an additional level of security because no communication between devices belonging to different VRFs is allowed unless it is explicitly configured. The IPv6 Multicast VRF Lite feature simplifies the management and troubleshooting of traffic belonging to a specific VRF.

IPv6 Multicast Process Switching and Fast Switching

A unified MFIB is used to provide both fast switching and process switching support for PIM-SM and PIM-SSM in IPv6 multicast. In process switching, the switch must examine, rewrite, and forward each packet. The packet is first received and copied into the system memory. The switch then looks up the Layer 3 network address in the routing table. The Layer 2 frame is then rewritten with the next-hop destination address and sent to the outgoing interface. The switch also computes the cyclic redundancy check (CRC). This switching method is the least scalable method for switching IPv6 packets.

IPv6 multicast fast switching allows switches to provide better packet forwarding performance than process switching. Information conventionally stored in a route cache is stored in several data structures for IPv6 multicast switching. The data structures provide optimized lookup for efficient packet forwarding.

In IPv6 multicast forwarding, the first packet is fast-switched if the PIM protocol logic allows it. In IPv6 multicast fast switching, the MAC encapsulation header is precomputed. IPv6 multicast fast switching uses the MFIB to make IPv6 destination prefix-based switching decisions. In addition to the MFIB, IPv6 multicast fast switching uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all MFIB entries.

The adjacency table is populated as adjacencies are discovered. Each time an adjacency entry is created (such as through ARP), a link-layer header for that adjacent node is precomputed and stored in the adjacency table. Once a route is determined, it points to a next hop and corresponding adjacency entry. It is subsequently used for encapsulation during switching of packets.

A route might have several paths to a destination prefix, such as when a switch is configured for simultaneous load balancing and redundancy. For each resolved path, a pointer is added for the adjacency corresponding to the next-hop interface for that path. This mechanism is used for load balancing across several paths.

Multiprotocol BGP for the IPv6 Multicast Address Family

The multiprotocol BGP for the IPv6 multicast address family feature provides multicast BGP extensions for IPv6 and supports the same features and functionality as IPv4 BGP. IPv6 enhancements to multicast BGP include support for an IPv6 multicast address family and network layer reachability information (NLRI) and next hop (the next switch in the path to the destination) attributes that use IPv6 addresses.

Multicast BGP is an enhanced BGP that allows the deployment of interdomain IPv6 multicast. Multiprotocol BGP carries routing information for multiple network layer protocol address families; for example, IPv6 address family and for IPv6 multicast routes. The IPv6 multicast address family contains routes used for RPF lookup by the IPv6 PIM protocol, and multicast BGP IPv6 provides for interdomain transport of the same. Users must use multiprotocol BGP for IPv6 multicast when using IPv6 multicast with BGP because the unicast BGP learned routes will not be used for IPv6 multicast.

Multicast BGP functionality is provided through a separate address family context. A subsequent address family identifier (SAFI) provides information about the type of the network layer reachability information that is carried in the attribute. Multiprotocol BGP unicast uses SAFI 1 messages, and multiprotocol BGP multicast uses SAFI 2 messages. SAFI 1 messages indicate that the routes are only usable for IP unicast, but not IP multicast. Because of this functionality, BGP routes in the IPv6 unicast RIB must be ignored in the IPv6 multicast RPF lookup.

A separate BGP routing table is maintained to configure incongruent policies and topologies (for example, IPv6 unicast and multicast) by using IPv6 multicast RPF lookup. Multicast RPF lookup is very similar to the IP unicast route lookup.

No MRIB is associated with the IPv6 multicast BGP table. However, IPv6 multicast BGP operates on the unicast IPv6 RIB when needed. Multicast BGP does not insert or update routes into the IPv6 unicast RIB.

NSF and SSO Support In IPv6 Multicast

Support for nonstop forwarding (NSF) and stateful switchover (SSO) is provided in IPv6 Multicast.

Bandwidth-Based CAC for IPv6 Multicast

The bandwidth-based call admission control (CAC) for IPv6 multicast feature implements a way to count per-interface mroute state limiters using cost multipliers. This feature can be used to provide bandwidth-based CAC on a per-interface basis in network environments where the multicast flows use different amounts of bandwidth.

This feature limits and accounts for IPv6 multicast state in detail. When this feature is configured, interfaces can be limited to the number of times they may be used as incoming or outgoing interfaces in the IPv6 multicast PIM topology.

With this feature, switch administrators can configure global limit cost commands for state matching access lists and specify which cost multiplier to use when accounting such state against the interface limits. This feature provides the required flexibility to implement bandwidth-based local CAC policy by tuning appropriate cost multipliers for different bandwidth requirements.

Implementing IPv6 Multicast

Enabling IPv6 Multicast Routing

To enable IPv6 multicast routing, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal	Enter global configuration mode.
Step 3	ipv6 multicast-routing Example: Switch(config)# ipv6 multicast-routing	Enables multicast routing on all IPv6-enabled interfaces and enables multicast forwarding for PIM and MLD on all enabled interfaces of the switch.
Step 4	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Customizing and Verifying the MLD Protocol

Customizing and Verifying MLD on an Interface

To customize and verify MLD on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface GigabitEthernet 1/0/1	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 4	ipv6 mld join-group [<i>group-address</i>] [include exclude] { <i>source-address</i> source-list [<i>acl</i>]} Example: Switch(config-if)# ipv6 mld join-group FF04::10	Configures MLD reporting for a specified group and source.
Step 5	ipv6 mld access-group <i>access-list-name</i> Example: Switch(config-if)# ipv6 access-list acc-grp-1	Allows the user to perform IPv6 multicast receiver access control.
Step 6	ipv6 mld static-group [<i>group-address</i>] [include exclude] { <i>source-address</i> source-list [<i>acl</i>]} Example: Switch(config-if)# ipv6 mld static-group ff04::10 include 100::1	Statically forwards traffic for the multicast group onto a specified interface and cause the interface to behave as if a MLD joiner were present on the interface.
Step 7	ipv6 mld query-max-response-time <i>seconds</i> Example: Switch(config-if)# ipv6 mld query-timeout 130	Configures the timeout value before the switch takes over as the querier for the interface.
Step 8	exit Example: Switch(config-if)# exit	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 9	show ipv6 mld groups [link-local] [<i>group-name</i> <i>group-address</i>] [<i>interface-type interface-number</i>] [detail explicit] Example:	Displays the multicast groups that are directly connected to the switch and that were learned through MLD.

	Command or Action	Purpose
	Switch# <code>show ipv6 mld groups GigabitEthernet 1/0/1</code>	
Step 10	show ipv6 mld groups summary Example: Switch# <code>show ipv6 mld groups summary</code>	Displays the number of (*, G) and (S, G) membership reports present in the MLD cache.
Step 11	show ipv6 mld interface [<i>type number</i>] Example: Switch# <code>show ipv6 mld interface GigabitEthernet 1/0/1</code>	Displays multicast-related information about an interface.
Step 12	debug ipv6 mld [<i>group-name</i> <i>group-address</i> <i>interface-type</i>] Example: Switch# <code>debug ipv6 mld</code>	Enables debugging on MLD protocol activity.
Step 13	debug ipv6 mld explicit [<i>group-name</i> <i>group-address</i>] Example: Switch# <code>debug ipv6 mld explicit</code>	Displays information related to the explicit tracking of hosts.
Step 14	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Implementing MLD Group Limits

Per-interface and global MLD limits operate independently of each other. Both per-interface and global MLD limits can be configured on the same switch. The number of MLD limits, globally or per interface, is not configured by default; the limits must be configured by the user. A membership report that exceeds either the per-interface or the global state limit is ignored.

Implementing MLD Group Limits Globally

To implement MLD group limits globally, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 mld** [*vrf vrf-name*] **state-limit number**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 mld [vrf vrf-name] state-limit number Example: Switch(config)# ipv6 mld state-limit 300	Limits the number of MLD states globally.
Step 4	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Implementing MLD Group Limits per Interface

To implement MLD group limits per interface, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface type number**
4. **ipv6 mld limit number [except]access-list**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface type number Example:	Specifies an interface type and number, and places the switch in interface configuration mode.

	Command or Action	Purpose
	<code>Switch(config)# interface GigabitEthernet 1/0/1</code>	
Step 4	ipv6 mld limit <i>number</i> [except] <i>access-list</i> Example: <code>Switch(config-if)# ipv6 mld limit 100</code>	Limits the number of MLD states on a per-interface basis.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring Explicit Tracking of Receivers to Track Host Behavior

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network and enables the fast leave mechanism to be used with MLD version 2 host reports.

To configuring explicit tracking of receivers to track host behavior, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <code>Switch> enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal	Enter global configuration mode.
Step 3	interface <i>type number</i> Example: <code>Switch(config)# interface GigabitEthernet 1/0/1</code>	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 4	ipv6 mld explicit-tracking <i>access-list-name</i> Example: <code>Switch(config-if)# ipv6 mld explicit-tracking list1</code>	Enables explicit tracking of hosts.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring Multicast User Authentication and Profile Support

Before you configure multicast user authentication and profile support, you should be aware of the following restrictions:

- The port, interface, VC, or VLAN ID is the user or subscriber identity. User identity by hostname, user ID, or password is not supported
- Enabling AAA Access Control for IPv6 Multicast

- Specifying Method Lists and Enabling Multicast Accounting
- Disabling the Switch from Receiving Unauthenticated Multicast Traffic Disabling the Switch from Receiving Unauthenticated Multicast Traffic
- Resetting Authorization Status on an MLD Interface

Enabling AAA Access Control for IPv6 Multicast

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	aaa new-model Example: Switch(config)# aaa new-model	Enables the AAA access control system.
Step 3	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Specifying Method Lists and Enabling Multicast Accounting

Perform this task to specify the method lists used for AAA authorization and accounting and how to enable multicast accounting on specified groups or channels on an interface.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	aaa authorization multicast default [<i>method3</i> <i>method4</i>] Example: Switch (config)# aaa authorization multicast default	Enables AAA authorization and sets parameters that restrict user access to an IPv6 multicast network.
Step 3	aaa accounting multicast default [start-stop stop-only] [broadcast] [<i>method1</i>] [<i>method2</i>] [<i>method3</i>] [<i>method2</i>] Example: Switch (config)# aaa accounting multicast default	Enables AAA accounting of IPv6 multicast services for billing or security purposes when you use RADIUS.
Step 4	interface <i>type number</i> Example: Switch (config)# interface FastEthernet 1/0	Specifies an interface type and number, and places the switch in interface configuration mode.

	Command or Action	Purpose
Step 5	ipv6 multicast aaa account receive access-list-name <i>access-list-name</i> [<i>throttle</i> <i>throttle-number</i>] Example: <pre>Switch (config-if)# ipv6 multicast aaa account receive list1</pre>	Enables AAA accounting on specified groups or channels.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Disabling the Switch from Receiving Unauthenticated Multicast Traffic

In some situations, access control may be needed to prevent multicast traffic from being received unless the subscriber is authenticated and the channels are authorized as per access control profiles. That is, there should be no traffic at all unless specified otherwise by access control profiles.

Perform this task to disable the switch from receiving multicast traffic to be received from unauthenticated groups or unauthorized channels.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	ipv6 multicast [vrfvrf-name] group-range <i>[access-list-name]</i> Example: <pre>Switch (config)# ipv6 multicast group-range</pre>	Disables multicast protocol actions and traffic forwarding for unauthorized groups or channels on all the interfaces in a switch.
Step 3	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Enabling MLD Proxy in IPv6

Beginning in privileged EXEC mode, follow these steps.

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	ipv6 mld host-proxy [group-acl] Example: <pre>Switch (config)# ipv6 mld host-proxy proxy-group</pre>	Enables the MLD proxy feature.
Step 3	ipv6 mld host-proxy interface [group-acl] Example:	Enables the MLD proxy feature on a specified interface on an RP.

Resetting Authorization Status on an MLD Interface

	Command or Action	Purpose
	Switch (config)# ipv6 mld host-proxy interface Ethernet 0/0	
Step 4	show ipv6 mld host-proxy [<i>interface-type interface-number</i>] group [<i>group-address</i>]] Example: Switch (config)# show ipv6 mld host-proxy Ethernet0/0	Displays IPv6 MLD host proxy information.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Resetting Authorization Status on an MLD Interface

If no interface is specified, authorization is reset on all MLD interfaces.

Beginning in privileged EXEC mode, follow these steps.

Procedure

	Command or Action	Purpose
Step 1	clear ipv6 multicast aaa authorization [<i>interface-type interface-number</i>]] Example: Switch # clear ipv6 multicast aaa authorization FastEthernet 1/0	Enter global configuration mode.
Step 2	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Resetting the MLD Traffic Counters

To reset the MLD traffic counters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	clear ipv6 mld traffic Example: Switch# <code>clear ipv6 mld traffic</code>	Resets all MLD traffic counters.
Step 4	show ipv6 mld traffic Example: Switch# <code>show ipv6 mld traffic</code>	Displays the MLD traffic counters.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Clearing the MLD Interface Counters

To clearing the MLD interface counters, perform this procedure

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	clear ipv6 mld counters <i>interface-type</i> Example: Switch# <code>clear ipv6 mld counters Ethernet1/0</code>	Clears the MLD interface counters.
Step 4	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring PIM

This section explains how to configure PIM.

Configuring PIM-SM and Displaying PIM-SM Information for a Group Range

To configuring PIM-SM and view PIM-SM information for a group range, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 pim rp-address <i>ipv6-address</i> [<i>group-access-list</i>] Example: Switch(config)# ipv6 pim rp-address 2001:DB8::01:800:200E:8C6C acc-grp-1	Configures the address of a PIM RP for a particular group range.
Step 4	exit Example: Switch(config)# exit	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 5	show ipv6 pim interface [<i>state-on</i>] [<i>state-off</i>] [<i>type-number</i>] Example: Switch# show ipv6 pim interface	Displays information about interfaces configured for PIM.
Step 6	show ipv6 pim group-map [<i>group-name</i> <i>group-address</i>] [<i>group-range</i> <i>group-mask</i>] [<i>info-source</i> { bsr default embedded-rp static }] Example: Switch# show ipv6 pim group-map	Displays an IPv6 multicast group mapping table.
Step 7	show ipv6 pim neighbor [detail] [<i>interface-type</i> <i>interface-number</i> count] Example: Switch# show ipv6 pim neighbor	Displays the PIM neighbors discovered by the Cisco IOS software.
Step 8	show ipv6 pim range-list [config] [<i>rp-address</i> <i>rp-name</i>] Example: Switch# show ipv6 pim range-list	Displays information about IPv6 multicast range lists.

	Command or Action	Purpose
Step 9	show ipv6 pim tunnel [<i>interface-type interface-number</i>] Example: Switch# show ipv6 pim tunnel	Displays information about the PIM register encapsulation and de-encapsulation tunnels on an interface.
Step 10	debug ipv6 pim [<i>group-name group-address interface interface-type bsr group mvpn neighbor</i>] Example: Switch# debug ipv6 pim	Enables debugging on PIM protocol activity.
Step 11	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring PIM Options

To configure PIM options, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 pim spt-threshold infinity [group-list access-list-name] Example: Switch(config)# ipv6 pim spt-threshold infinity group-list acc-grp-1	Configures when a PIM leaf switch joins the SPT for the specified groups.
Step 4	ipv6 pim accept-register { list access-list route-map map-name } Example: Switch(config)# ipv6 pim accept-register route-map reg-filter	Accepts or rejects registers at the RP.
Step 5	interface <i>type number</i> Example:	Specifies an interface type and number, and places the switch in interface configuration mode.

	Command or Action	Purpose
	Switch(config)# interface GigabitEthernet 1/0/1	
Step 6	ipv6 pim dr-priority <i>value</i> Example: Switch(config-if)# ipv6 pim dr-priority 3	Configures the DR priority on a PIM switch.
Step 7	ipv6 pim hello-interval <i>seconds</i> Example: Switch(config-if)# ipv6 pim hello-interval 45	Configures the frequency of PIM hello messages on an interface.
Step 8	ipv6 pim join-prune-interval <i>seconds</i> Example: Switch(config-if)# ipv6 pim join-prune-interval 75	Configures periodic join and prune announcement intervals for a specified interface.
Step 9	exit Example: Switch(config-if)# exit	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 10	ipv6 pim join-prune statistic [<i>interface-type</i>] Example: Switch(config-if)# show ipv6 pim join-prune statistic	Displays the average join-prune aggregation for the most recently aggregated packets for each interface.
Step 11	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring Bidirectional PIM and Displaying Bidirectional PIM Information

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	ipv6 pim [<i>vrf vrf-name</i>] rp-address <i>ipv6-address</i> [<i>group-access-list</i>] [bidir] Example: Switch (config) # ipv6 pim rp-address 2001:DB8::01:800:200E:8C6C bidir	Configures the address of a PIM RP for a particular group range. Use of the bidir keyword means that the group range will be used for bidirectional shared-tree forwarding.

	Command or Action	Purpose
Step 3	exit Example: <pre>Switch (config-if) # exit</pre>	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 4	show ipv6 pim [vrf vrf-name] df [interface-type interface-number] [rp-address] Example: <pre>Switch (config) # show ipv6 pim df</pre>	Displays the designated forwarder (DF)-election state of each interface for RP.
Step 5	show ipv6 pim [vrf vrf-name] df winner [interface-type interface-number] [rp-address] Example: <pre>Switch (config-if) # show ipv6 pim df winner ethernet 1/0 200::1</pre>	Displays the DF-election winner on each interface for each RP.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Resetting the PIM Traffic Counters

If PIM malfunctions or in order to verify that the expected number of PIM packets are received and sent, the user can clear PIM traffic counters. Once the traffic counters are cleared, the user can enter the `show ipv6 pim` traffic command to verify that PIM is functioning correctly and that PIM packets are being received and sent correctly.

To resetting the PIM traffic counters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	clear ipv6 pim traffic Example: <pre>Switch# clear ipv6 pim traffic</pre>	Resets the PIM traffic counters.

	Command or Action	Purpose
Step 4	show ipv6 pim traffic Example: Switch# <code>show ipv6 pim traffic</code>	Displays the PIM traffic counters.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Clearing the PIM Topology Table to Reset the MRIB Connection

No configuration is necessary to use the MRIB. However, users may in certain situations want to clear the PIM topology table in order to reset the MRIB connection and verify MRIB information.

To clear the PIM topology table to reset the MRIB connection, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	clear ipv6 pim topology [<i>group-name</i> <i>group-address</i>] Example: Switch# <code>clear ipv6 pim topology FF04::10</code>	Clears the PIM topology table.
Step 4	show ipv6 mrib client [<i>filter</i>] [<i>name</i> { <i>client-name</i> <i>client-name</i> : <i>client-id</i> }] Example: Switch# <code>show ipv6 mrib client</code>	Displays multicast-related information about an interface.
Step 5	show ipv6 mrib route { <i>link-local</i> <i>summary</i> [<i>sourceaddress-or-name</i> *] [<i>groupname-or-address</i> [<i>prefix-length</i>]]] Example: SwitchSwitch# <code>show ipv6 mrib route</code>	Displays the MRIB route information.

	Command or Action	Purpose
Step 6	show ipv6 pim topology [<i>groupname-or-address</i> [<i>sourceaddress-or-name</i>] link-local route-count [detail]] Example: Switch# <code>show ipv6 pim topology</code>	Displays PIM topology table information for a specific group or all groups.
Step 7	debug ipv6 mrib client Example: Switch# <code>debug ipv6 mrib client</code>	Enables debugging on MRIB client management activity.
Step 8	debug ipv6 mrib io Example: Switch# <code>debug ipv6 mrib io</code>	Enables debugging on MRIB I/O events.
Step 9	debug ipv6 mrib proxy Example: Switch# <code>debug ipv6 mrib proxy</code>	Enables debugging on MRIB proxy activity between the switch processor and line cards on distributed switch platforms.
Step 10	debug ipv6 mrib route [<i>group-name</i> <i>group-address</i>] Example: Switch# <code>debug ipv6 mrib route</code>	Displays information about MRIB routing entry-related activity.
Step 11	debug ipv6 mrib table Example: Switch# <code>debug ipv6 mrib table</code>	Enables debugging on MRIB table management activity.
Step 12	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring a BSR

The tasks included here are described below.

Configuring a BSR and Verifying BSR Information

To configure and verify BSR Information, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> enable	Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 pim bsr candidate bsr <i>ipv6-address[hash-mask-length] [priority priority-value]</i> Example: Switch(config)# ipv6 pim bsr candidate bsr 2001:DB8:3000:3000::42 124 priority 10	Configures a switch to be a candidate BSR.
Step 4	interface type number Example: Switch(config)# interface GigabitEthernet 1/0/1	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 5	ipv6 pim bsr border Example: Switch(config-if)# ipv6 pim bsr border	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 6	exit Example: Switch(config-if)# exit	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 7	show ipv6 pim bsr {election rp-cache candidate-rp} Example: Switch(config-if)# show ipv6 pim bsr election	Displays information related to PIM BSR protocol processing.
Step 8	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Sending PIM RP Advertisements to the BSR

To sending PIM RP advertisements to the BSR, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> enable	Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ipv6 pim bsr candidate rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i>] [interval <i>seconds</i>] Example: Switch(config)# ipv6 pim bsr candidate rp 2001:DB8:3000:3000::42 priority 0	Sends PIM RP advertisements to the BSR.
Step 4	interface <i>type number</i> Example: Switch(config)# interface GigabitEthernet 1/0/1	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 5	ipv6 pim bsr border Example: Switch(config-if)# ipv6 pim bsr border	Configures a border for all BSMs of any scope on a specified interface.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring BSR for Use Within Scoped Zones

To configure BSR for use within scoped zones, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ipv6 pim bsr candidate rp <i>ipv6-address</i> <i>[hash-mask-length] [priority priority-value]</i> Example: <pre>Switch(config)# ipv6 pim bsr candidate bsr 2001:DB8:1:1:4</pre>	Configures a switch to be a candidate BSR.
Step 4	ipv6 pim bsr candidate rp <i>ipv6-address</i> [group-list <i>access-list-name]</i> [priority priority-value] [interval <i>seconds]</i> Example: <pre>Switch(config)# ipv6 pim bsr candidate rp 2001:DB8:1:1:1 group-list list scope 6</pre>	Configures the candidate RP to send PIM RP advertisements to the BSR.
Step 5	interface <i>type number</i> Example: <pre>Switch(config-if)# interface GigabitEthernet 1/0/1</pre>	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 6	ipv6 multicast boundary scope <i>scope-value</i> Example: <pre>Switch(config-if)# ipv6 multicast boundary scope 6</pre>	Configures a multicast boundary on the interface for a specified scope.
Step 7	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring BSR Switches to Announce Scope-to-RP Mappings

IPv6 BSR switches can be statically configured to announce scope-to-RP mappings directly instead of learning them from candidate-RP messages. A user might want to configure a BSR switch to announce scope-to-RP mappings so that an RP that does not support BSR is imported into the BSR. Enabling this feature also allows an RP positioned outside the enterprise's BSR domain to be learned by the known remote RP on the local candidate BSR switch.

To configure BSR switches to announce Scope-to-RP mappings, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ipv6 pim bsr announced rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i>] Example: <pre>Switch(config)# ipv6 pim bsr announced rp 2001:DB8:3000:3000::42 priority 0</pre>	Announces scope-to-RP mappings directly from the BSR for the specified candidate RP.
Step 4	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring SSM Mapping

When the SSM mapping feature is enabled, DNS-based SSM mapping is automatically enabled, which means that the switch will look up the source of a multicast MLD version 1 report from a DNS server.

You can use either DNS-based or static SSM mapping, depending on your switch configuration. If you choose to use static SSM mapping, you can configure multiple static SSM mappings. If multiple static SSM mappings are configured, the source addresses of all matching access lists will be used.



Note To use DNS-based SSM mapping, the switch needs to find at least one correctly configured DNS server, to which the switch may be directly attached.

To configuring SSM mapping, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ipv6 mld ssm-map enable Example:	Enables the SSM mapping feature for groups in the configured SSM range.

	Command or Action	Purpose
	Switch(config)# ipv6 mld ssm-map enable	
Step 4	no ipv6 mld ssm-map query dns Example: Switch(config)# no ipv6 mld ssm-map query dns	Disables DNS-based SSM mapping.
Step 5	ipv6 mld ssm-map static access-list source-address Example: Switch(config-if)# ipv6 mld ssm-map static SSM_MAP_ACL_2 2001:DB8:1::1	Configures static SSM mappings.
Step 6	exit Example: Switch(config-if)# exit	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 7	show ipv6 mld ssm-map [source-address] Example: Switch(config-if)# show ipv6 mld ssm-map	Displays SSM mapping information.
Step 8	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring Static Mroutes

Static multicast routes (mroutes) in IPv6 can be implemented as an extension of IPv6 static routes. You can configure your switch to use a static route for unicast routing only, to use a static multicast route for multicast RPF selection only, or to use a static route for both unicast routing and multicast RPF selection.

To configure static mroutes, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ipv6 route { <i>ipv6-prefix / prefix-length ipv6-address interface-type interface-number ipv6-address</i> } [<i>administrative-distance</i>] [<i>administrative-multicast-distance</i>] <i>unicast</i> <i>multicast</i>] [tag tag] Example: Switch(config)# ipv6 route 2001:DB8::/64 6::6 100	Establishes static IPv6 routes. The example shows a static route used for both unicast routing and multicast RPF selection.
Step 4	exit Example: Switch# exit	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 5	show ipv6 mroute [<i>link-local</i> [<i>group-name</i> <i>group-address</i> [<i>source-address</i> <i>source-name</i>]]] [summary] [count] Example: Switch# show ipv6 mroute ff07::1	Displays the contents of the IPv6 multicast routing table.
Step 6	show ipv6 mroute [<i>link-local</i> <i>group-name</i> <i>group-address</i>] active [<i>kpbs</i>] Example: Switch(config-if)# show ipv6 mroute active	Displays the active multicast streams on the switch.
Step 7	show ipv6 rpf [<i>ipv6-prefix</i>] Example: Switch(config-if)# show ipv6 rpf 2001::1:1:2	Checks RPF information for a given unicast host address and prefix.
Step 8	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Using MFIB in IPv6 Multicast

Multicast forwarding is automatically enabled when IPv6 multicast routing is enabled.

Verifying MFIB Operation in IPv6 Multicast

To verify MFIB operation in IPv6 multicast

Procedure

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	show ipv6 mfib [link-local verbose <i>group-address-name</i> <i>ipv6-prefix / prefix-length</i> <i>source-address-name</i> count interface status summary] Example: Switch# show ipv6 mfib	Displays the forwarding entries and interfaces in the IPv6 MFIB.
Step 3	show ipv6 mfib [all linkscope <i>group-name</i> <i>group-address</i> [<i>source-name</i> <i>source-address</i>]] count Example: Switch# show ipv6 mfib ff07::1	Displays the contents of the IPv6 multicast routing table.
Step 4	show ipv6 mfib interface Example: Switch# show ipv6 mfib interface	Displays information about IPv6 multicast-enabled interfaces and their forwarding status.
Step 5	show ipv6 mfib status Example: Switch# show ipv6 mfib status	Displays general MFIB configuration and operational status.
Step 6	show ipv6 mfib summary Example: Switch# show ipv6 mfib summary	Displays summary information about the number of IPv6 MFIB entries and interfaces.
Step 7	debug ipv6 mfib [<i>group-name</i> <i>group-address</i>] [adjacency db fs init interface mrrib [detail] nat pak platform ppr ps signal table] Example: Switch# debug ipv6 mfib FF04::10 pak	Enables debugging output on the IPv6 MFIB.

Resetting MFIB Traffic Counters

To reset MFIB traffic counters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Switch> <code>enable</code>	
Step 2	clear ipv6 mfib counters [<i>group-name</i> group-address [<i>source-address</i> <i>source-name</i>]] Example: Switch# <code>clear ipv6 mfib counters FF04::10</code>	Resets all active MFIB traffic counters.



PART **IV**

Layer 2

- [Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling, on page 393](#)
- [Configuring Spanning Tree Protocol, on page 417](#)
- [Configuring Multiple Spanning-Tree Protocol, on page 441](#)
- [Configuring Optional Spanning-Tree Features, on page 481](#)
- [Configuring Bidirection Forwarding Detection, on page 513](#)
- [Configuring EtherChannels, on page 541](#)
- [Configuring Link-State Tracking, on page 571](#)
- [Configuring Resilient Ethernet Protocol, on page 577](#)
- [Configuring Flex Links and the MAC Address-Table Move Update Feature, on page 595](#)
- [Configuring UniDirectional Link Detection, on page 609](#)



CHAPTER 23

Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling

- [Finding Feature Information, on page 393](#)
- [Prerequisites for Configuring Tunneling, on page 393](#)
- [Information about Tunneling, on page 395](#)
- [How to Configure Tunneling, on page 403](#)
- [Configuration Examples for IEEE 802.1Q and Layer 2 Protocol Tunneling, on page 414](#)
- [Monitoring Tunneling Status, on page 416](#)
- [Where to Go Next, on page 416](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Configuring Tunneling

The following sections list prerequisites and considerations for configuring IEEE 802.1Q and Layer 2 protocol tunneling.

IEEE 802.1Q Tunneling

Although IEEE 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities between some Layer 2 features and Layer 3 switching.

- A tunnel port cannot be a routed port.
- IP routing is not supported on a VLAN that includes IEEE 802.1Q tunnel ports. Packets received from a tunnel port are forwarded based only on Layer 2 information. If routing is enabled on a device virtual interface (SVI) that includes tunnel ports, untagged IP packets received from the tunnel port are recognized

and routed by the device. Customers can access the Internet through its native VLAN. If this access is not needed, you should not configure SVIs on VLANs that include tunnel ports.

- Fallback bridging is not supported on tunnel ports. Because all IEEE 802.1Q-tagged packets received from a tunnel port are treated as non-IP packets, if fallback bridging is enabled on VLANs that have tunnel ports configured, IP packets would be improperly bridged across VLANs. Therefore, you must not enable fallback bridging on VLANs with tunnel ports.
- Tunnel ports do not support IP access control lists (ACLs).
- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), and UniDirectional Link Detection (UDLD) are supported on IEEE 802.1Q tunnel ports.
- Dynamic Trunking Protocol (DTP) is not compatible with IEEE 802.1Q tunneling because you must manually configure asymmetric links with tunnel ports and trunk ports.
- VLAN Trunking Protocol (VTP) does not work between devices that are connected by an asymmetrical link or devices that communicate through a tunnel.
- Loopback detection is supported on IEEE 802.1Q tunnel ports.
- When a port is configured as an IEEE 802.1Q tunnel port, spanning-tree bridge protocol data unit (BPDU) filtering is automatically enabled on the interface. Cisco Discovery Protocol (CDP) and the Layer Link Discovery Protocol (LLDP) are automatically disabled on the interface.

Layer 2 Protocol Tunneling

- The device supports tunneling of CDP, STP, including multiple STP (MSTP), and VTP. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or access ports.
- The device does not support Layer 2 protocol tunneling on ports with switchport mode dynamic auto or dynamic desirable.
- DTP is not compatible with layer 2 protocol tunneling.
- The edge devices on the outbound side of the service-provider network restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel and access ports in the same metro VLAN.
- For interoperability with third-party vendor devices, the device supports a Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor devices that have different ways of controlling protocol tunneling. When Layer 2 protocol tunneling is enabled on ingress ports on a device, egress trunk ports forward the tunneled packets with a special encapsulation. If you also enable Layer 2 protocol tunneling on the egress trunk port, this behavior is bypassed, and the device forwards control PDUs without any processing or modification.

- The device supports PAgP, LACP, and UDLD tunneling for emulated point-to-point network topologies. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on access ports.
- If you enable PAgP or LACP tunneling, we recommend that you also enable UDLD on the interface for faster link-failure detection.
- Loopback detection is not supported on Layer 2 protocol tunneling of PAgP, LACP, or UDLD packets.
- EtherChannel port groups are compatible with tunnel ports when the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or an access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops. The port also shuts down when a configured shutdown threshold for the protocol is reached. You can manually reenabling the port (by entering a **shutdown** and a **no shutdown** command sequence). If errdisable recovery is enabled, the operation is retried after a specified time interval.
- Only decapsulated PDUs are forwarded to the customer network. The spanning-tree instance running on the service-provider network does not forward BPDUs to tunnel ports. CDP packets are not forwarded from tunnel ports.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, shutdown threshold for the PDUs generated by the customer network. If the limit is exceeded, the port shuts down. You can also limit BPDU rate by using QoS ACLs and policy maps on a tunnel port.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, drop threshold for the PDUs generated by the customer network. If the limit is exceeded, the port drops PDUs until the rate at which it receives them is below the drop threshold.
- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites so that the customer virtual network operates properly, you can give PDUs higher priority within the service-provider network than data packets received from the same tunnel port. By default, the PDUs use the same CoS value as data packets.

Layer 2 Tunneling for EtherChannels

To configure Layer 2 point-to-point tunneling to facilitate the automatic creation of EtherChannels, you need to configure both the SP (service-provider) edge switch and the customer device.

Information about Tunneling

IEEE 802.1Q and Layer 2 Protocol Overview

Virtual private networks (VPNs) provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and are required to maintain the VLAN and Layer 2 protocol configurations of each customer without impacting the traffic of other customers.



Note IEEE 802.1Q and Layer 2 protocol tunneling are supported only on Cisco Catalyst 3560-CX switches.

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

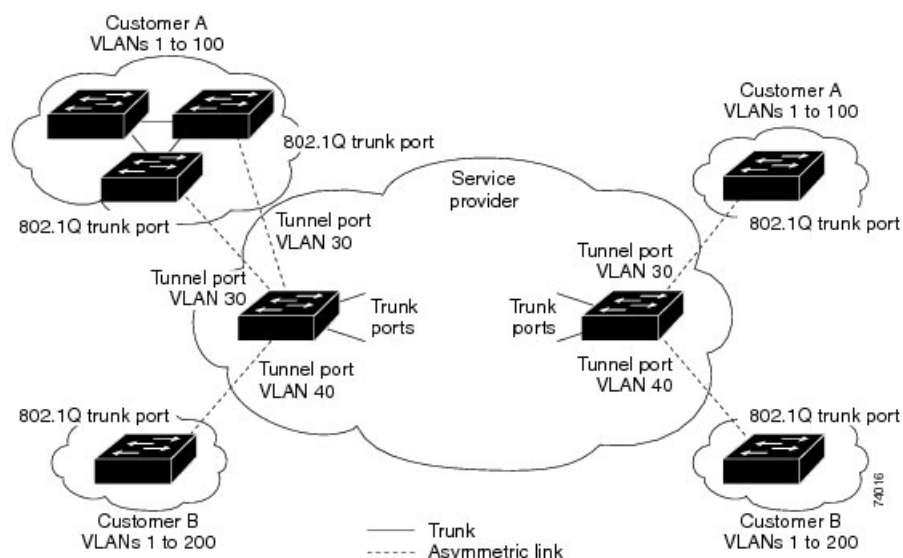
IEEE 802.1Q Tunneling

Business customers of service providers often have specific requirements for VLAN IDs and the number of VLANs to be supported. The VLAN ranges required by different customers in the same service-provider network might overlap, and traffic of customers through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the VLAN limit (4096) of the IEEE 802.1Q specification.

Using the IEEE 802.1Q tunneling feature, service providers can use a single VLAN to support customers who have multiple VLANs. Customer VLAN IDs are preserved, and traffic from different customers is segregated within the service-provider network, even when they appear to be in the same VLAN. Using IEEE 802.1Q tunneling expands VLAN space by using a VLAN-in-VLAN hierarchy and retagging the tagged packets. A port configured to support IEEE 802.1Q tunneling is called a tunnel port. When you configure tunneling, you assign a tunnel port to a VLAN ID that is dedicated to tunneling. Each customer requires a separate service-provider VLAN ID, but that VLAN ID supports all of the customer's VLANs.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an IEEE 802.1Q trunk port on the customer device and into a tunnel port on the service-provider edge device. The link between the customer device and the edge device is asymmetric because one end is configured as an IEEE 802.1Q trunk port, and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID that is unique to each customer.

Figure 15: IEEE 802.1Q Tunnel Ports in a Service-Provider Network



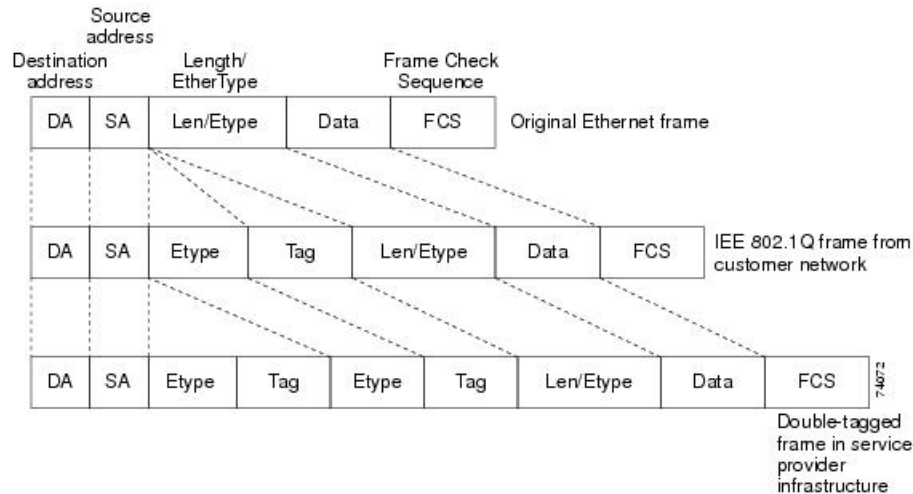
Packets coming from the customer trunk port into the tunnel port on the service-provider edge device are normally IEEE 802.1Q-tagged with the appropriate VLAN ID. The tagged packets remain intact inside the device and when they exit the trunk port into the service-provider network, they are encapsulated with another

layer of an IEEE 802.1Q tag (called the metro tag) that contains the VLAN ID that is unique to the customer. The original customer IEEE 802.1Q tag is preserved in the encapsulated packet. Therefore, packets entering the service-provider network are double-tagged, with the outer (metro) tag containing the customer's access VLAN ID, and the inner VLAN ID being that of the incoming traffic.

When the double-tagged packet enters another trunk port in a service-provider core device, the outer tag is stripped as the device processes the packet. When the packet exits another trunk port on the same core device, the same metro tag is again added to the packet.

Figure 16: Original (Normal), IEEE 802.1Q, and Double-Tagged Ethernet Packet Formats

This figure shows the tag structures of the double-tagged packets.



When the packet enters the trunk port of the service-provider egress device, the outer tag is again stripped as the device internally processes the packet. However, the metro tag is not added when the packet is sent out the tunnel port on the edge device into the customer network. The packet is sent as a normal IEEE 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

In the above network figure, Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the edge device tunnel ports with IEEE 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original VLAN number, for example, VLAN 100. Even if both Customers A and B have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the outer tag is different. Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

At the outbound tunnel port, the original VLAN numbers on the customer's network are recovered. It is possible to have multiple levels of tunneling and tagging, but the device supports only one level in this release.

If traffic coming from a customer network is not tagged (native VLAN frames), these packets are bridged or routed as normal packets. All packets entering the service-provider network through a tunnel port on an edge device are treated as untagged packets, whether they are untagged or already tagged with IEEE 802.1Q headers. The packets are encapsulated with the metro tag VLAN ID (set to the access VLAN of the tunnel port) when they are sent through the service-provider network on an IEEE 802.1Q trunk port. The priority field on the metro tag is set to the interface class of service (CoS) priority configured on the tunnel port. (The default is zero if none is configured.)

IEEE 802.1Q Tunneling Configuration Guidelines

When you configure IEEE 802.1Q tunneling, you should always use an asymmetrical link between the customer device and the edge device, with the customer device port configured as an IEEE 802.1Q trunk port and the edge device port configured as a tunnel port.

Assign tunnel ports only to VLANs that are used for tunneling.

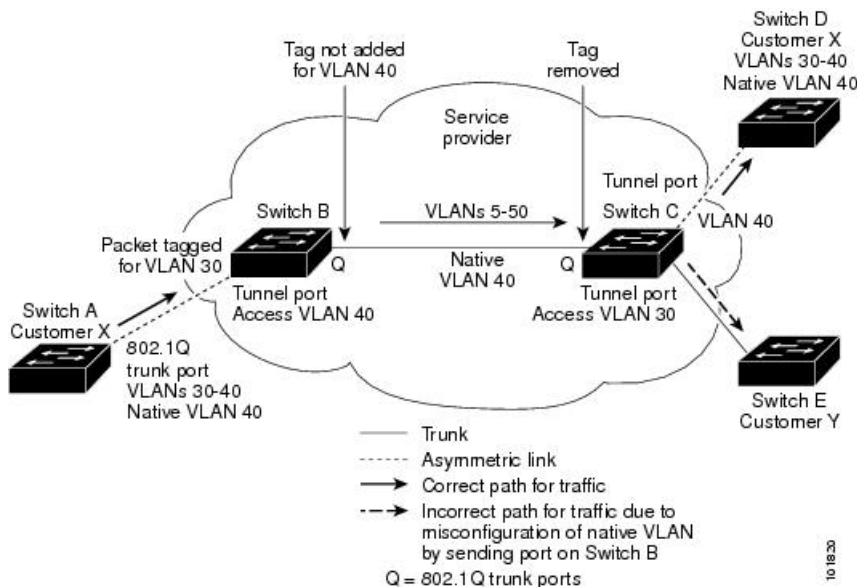
Configuration requirements for native VLANs and for and maximum transmission units (MTUs) are explained in these next sections.

Native VLANs

When configuring IEEE 802.1Q tunneling on an edge device, you must use IEEE 802.1Q trunk ports for sending packets into the service-provider network. However, packets going through the core of the service-provider network can be carried through IEEE 802.1Q trunks, ISL trunks, or nontrunking links. When IEEE 802.1Q trunks are used in these core devices, the native VLANs of the IEEE 802.1Q trunks must not match any native VLAN of the nontrunking (tunneling) port on the same device because traffic on the native VLAN would not be tagged on the IEEE 802.1Q sending trunk port.

In the following network figure, VLAN 40 is configured as the native VLAN for the IEEE 802.1Q trunk port from Customer X at the ingress edge device in the service-provider network (Device B). Device A of Customer X sends a tagged packet on VLAN 30 to the ingress tunnel port of Device B in the service-provider network, which belongs to access VLAN 40. Because the access VLAN of the tunnel port (VLAN 40) is the same as the native VLAN of the edge device trunk port (VLAN 40), the metro tag is not added to tagged packets received from the tunnel port. The packet carries only the VLAN 30 tag through the service-provider network to the trunk port of the egress-edgedevice (Device C) and is misdirected through the egress device tunnel port to Customer Y.

Figure 17: Potential Problems with IEEE 802.1Q Tunneling and Native VLANs



These are some ways to solve this problem:

•

- Use the **vlan dot1q tag native** global configuration command to configure the edge devices so that all packets going out an IEEE 802.1Q trunk, including the native VLAN, are tagged. If the device is configured to tag native VLAN packets on all IEEE 802.1Q trunks, the device accepts untagged packets, but sends only tagged packets.
- Ensure that the native VLAN ID on the edge devices trunk port is not within the customer VLAN range. For example, if the trunk port carries traffic of VLANs 100 to 200, assign the native VLAN a number outside that range.

System MTU

The default system MTU for traffic on the device is 1500 bytes.

You can configure 10-Gigabit and Gigabit Ethernet ports to support frames larger than 1500 bytes by using the **system mtu jumbo** global configuration command.

The system MTU and system jumbo MTU values do not include the IEEE 802.1Q header. Because the IEEE 802.1Q tunneling feature increases the frame size by 4 bytes when the metro tag is added, you must configure all devices in the service-provider network to be able to process maximum frames by adding 4 bytes to the system MTU and system jumbo MTU sizes.

For example, the device supports a maximum frame size of 1496 bytes with one of these configurations:

- The device has a system jumbo MTU value of 1500 bytes, and the **switchport mode dot1q tunnel** interface configuration command is configured on a 10-Gigabit or Gigabit Ethernet device port.
- The device member has a system MTU value of 1500 bytes, and the **switchport mode dot1q tunnel** interface configuration command is configured on a Fast Ethernet port of the member.

Default IEEE 802.1Q Tunneling Configuration

By default, IEEE 802.1Q tunneling is disabled because the default switchport mode is dynamic auto. Tagging of IEEE 802.1Q native VLAN packets on all IEEE 802.1Q trunk ports is also disabled.

Layer 2 Protocol Tunneling Overview

Customers at different sites connected across a service-provider network need to use various Layer 2 protocols to scale their topologies to include all remote sites, as well as the local sites. STP must run properly, and every VLAN should build a proper spanning tree that includes the local site and all remote sites across the service-provider network. Cisco Discovery Protocol (CDP) must discover neighboring Cisco devices from local and remote sites. VLAN Trunking Protocol (VTP) must provide consistent VLAN configuration throughout all sites in the customer network.

When protocol tunneling is enabled, edge devices on the inbound side of the service-provider network encapsulate Layer 2 protocol packets with a special MAC address and send them across the service-provider network. Core devices in the network do not process these packets but forward them as normal packets. Layer 2 protocol data units (PDUs) for CDP, STP, or VTP cross the service-provider network and are delivered to customer devices on the outbound side of the service-provider network. Identical packets are received by all customer ports on the same VLANs with these results:

- Users on each of a customer's sites can properly run STP, and every VLAN can build a correct spanning tree based on parameters from all sites and not just from the local site.

- CDP discovers and shows information about the other Cisco devices connected through the service-provider network.
- VTP provides consistent VLAN configuration throughout the customer network, propagating to all devices through the service provider.



Note To provide interoperability with third-party vendors, you can use the Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor devices that have different ways of controlling protocol tunneling. You implement bypass mode by enabling Layer 2 protocol tunneling on the egress trunk port. When Layer 2 protocol tunneling is enabled on the trunk port, the encapsulated tunnel MAC address is removed and the protocol packets have their normal MAC address.

Layer 2 protocol tunneling can be used independently or can enhance IEEE 802.1Q tunneling. If protocol tunneling is not enabled on IEEE 802.1Q tunneling ports, remote devices at the receiving end of the service-provider network do not receive the PDUs and cannot properly run STP, CDP, and VTP. When protocol tunneling is enabled, Layer 2 protocols within each customer's network are totally separate from those running within the service-provider network. Customer devices on different sites that send traffic through the service-provider network with IEEE 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If IEEE 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer device through access ports and by enabling tunneling on the service-provider access port.

For example, in the following figure (Layer 2 Protocol Tunneling), Customer X has four devices in the same VLAN, that are connected through the service-provider network. If the network does not tunnel PDUs, devices on the far ends of the network cannot properly run STP, CDP, and VTP. For example, STP for a VLAN on a device in Customer X, Site 1, will build a spanning tree on the devices at that site without considering convergence parameters based on Customer X's device in Site 2. This could result in the topology shown in the Layer 2 Network Topology without Proper Convergence figure.

Figure 18: Layer 2 Protocol Tunneling

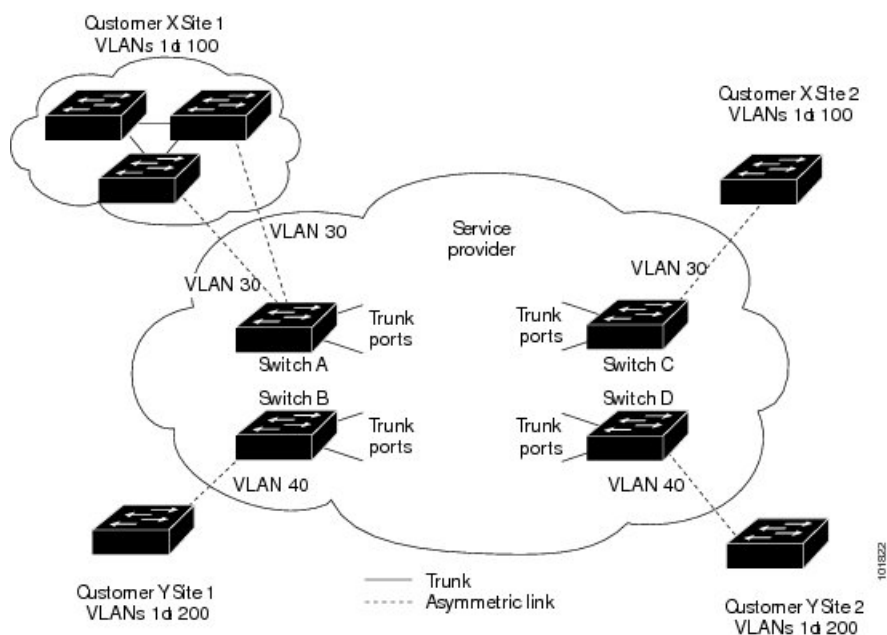
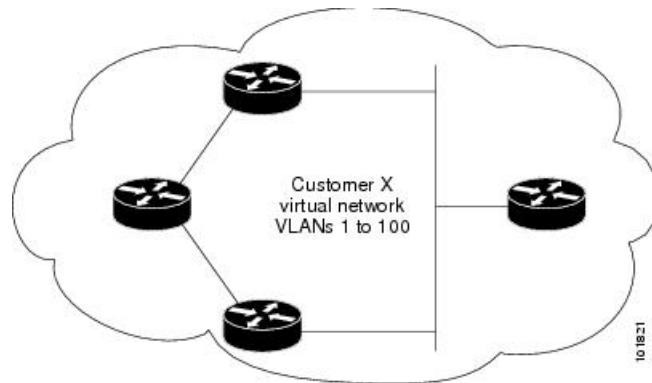


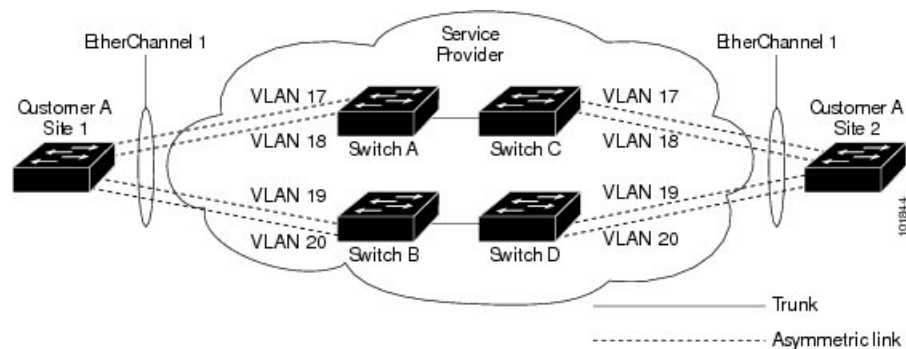
Figure 19: Layer 2 Network Topology Without Proper Convergence



In an SP network, you can use Layer 2 protocol tunneling to enhance the creation of EtherChannels by emulating a point-to-point network topology. When you enable protocol tunneling (PAGP or LACP) on the SP device, remote customer devices receive the PDUs and can negotiate the automatic creation of EtherChannels.

For example, in the following figure (Layer 2 Protocol Tunneling for EtherChannels), Customer A has two devices in the same VLAN that are connected through the SP network. When the network tunnels PDUs, devices on the far ends of the network can negotiate the automatic creation of EtherChannels without needing dedicated lines.

Figure 20: Layer 2 Protocol Tunneling for EtherChannels



Layer 2 Protocol Tunneling on Ports

You can enable Layer 2 protocol tunneling (by protocol) on the ports that are connected to the customer in the edge devices of the service-provider network. The service-provider edge devices connected to the customer device perform the tunneling process. Edge device tunnel ports are connected to customer IEEE 802.1Q trunk ports. Edge device access ports are connected to customer access ports. The edge devices connected to the customer device perform the tunneling process.

You can enable Layer 2 protocol tunneling on ports that are configured as access ports or tunnel ports. You cannot enable Layer 2 protocol tunneling on ports configured in either **switchport mode dynamic auto** mode (the default mode) or **switchport mode dynamic desirable** mode.

The device supports Layer 2 protocol tunneling for CDP, STP, and VTP. For emulated point-to-point network topologies, it also supports PAGP, LACP, and UDLD protocols. The device does not support Layer 2 protocol tunneling for LLDP.



Note PAgP, LACP, and UDLD protocol tunneling is only intended to emulate a point-to-point topology. An erroneous configuration that sends tunneled packets to many ports could lead to a network failure.

When the Layer 2 PDUs that entered the service-provider inbound edge device through a Layer 2 protocol-enabled port exit through the trunk port into the service-provider network, the device overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If IEEE 802.1Q tunneling is enabled, packets are also double-tagged; the outer tag is the customer metro tag, and the inner tag is the customer's VLAN tag. The core devices ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The edge devices on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel or access ports in the same metro VLAN. Therefore, the Layer 2 PDUs remain intact and are delivered across the service-provider infrastructure to the other side of the customer network.

See the Layer 2 Protocol Tunneling figure in [Layer 2 Protocol Tunneling Overview, on page 399](#), with Customer X and Customer Y in access VLANs 30 and 40, respectively. Asymmetric links connect the customers in Site 1 to edge devices in the service-provider network. The Layer 2 PDUs (for example, BPDUs) coming into Device B from Customer Y in Site 1 are forwarded to the infrastructure as double-tagged packets with the well-known MAC address as the destination MAC address. These double-tagged packets have the metro VLAN tag of 40, as well as an inner VLAN tag (for example, VLAN 100). When the double-tagged packets enter Device D, the outer VLAN tag 40 is removed, the well-known MAC address is replaced with the respective Layer 2 protocol MAC address, and the packet is sent to Customer Y on Site 2 as a single-tagged frame in VLAN 100.

You can also enable Layer 2 protocol tunneling on access ports on the edge device connected to access or trunk ports on the customer device. In this case, the encapsulation and decapsulation process is the same as described in the previous paragraph, except that the packets are not double-tagged in the service-provider network. The single tag is the customer-specific access VLAN tag.

Default Layer 2 Protocol Tunneling Configuration

The following table shows the default Layer 2 protocol tunneling configuration.

Table 39: Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled.
Shutdown threshold	None set.
Drop threshold	None set.
CoS Value	If a CoS value is configured on the interface, that value is used to set the BPDU CoS value for Layer 2 protocol tunneling. If no CoS value is configured at the interface level, the default value for CoS marking of L2 protocol tunneling BPDUs is 5. This does not apply to data traffic.

How to Configure Tunneling

Configuring an IEEE 802.1Q Tunneling Port

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport access vlan *vlan-id***
5. **switchport mode dot1q-tunnel**
6. **exit**
7. **vlan dot1q tag native**
8. **end**
9. Use one of the following:
 - **show dot1q-tunnel**
 - **show running-config interface**
10. **show vlan dot1q tag native**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	Enters interface configuration mode for the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer device. Valid interfaces include physical interfaces and port-channel logical interfaces (port channels 1 to 48).

	Command or Action	Purpose
Step 4	switchport access vlan <i>vlan-id</i> Example: Switch(config-if) # switchport access vlan 2	Specifies the default VLAN, which is used if the interface stops trunking. This VLAN ID is specific to the particular customer.
Step 5	switchport mode dot1q-tunnel Example: Switch(config-if) # switchport mode dot1q-tunnel	Sets the interface as an IEEE 802.1Q tunnel port. Note Use the no switchport mode dot1q-tunnel interface configuration command to return the port to the default state of dynamic desirable.
Step 6	exit Example: Switch(config-if) # exit	Returns to privileged EXEC mode.
Step 7	vlan dot1q tag native Example: Switch(config) # vlan dot1q tag native	(Optional) Sets the device to enable tagging of native VLAN packets on all IEEE 802.1Q trunk ports. When not set, and a customer VLAN ID is the same as the native VLAN, the trunk port does not apply a metro tag, and packets could be sent to the wrong destination. Note Use the no vlan dot1q tag native global configuration command to disable tagging of native VLAN packets.
Step 8	end Example: Switch(config) # end	Returns to privileged EXEC mode.
Step 9	Use one of the following: <ul style="list-style-type: none"> • show dot1q-tunnel • show running-config interface Example: Switch# show dot1q-tunnel or Switch# show running-config interface	Displays the ports configured for IEEE 802.1Q tunneling. Displays the ports that are in tunnel mode.
Step 10	show vlan dot1q tag native	Displays IEEE 802.1Q native VLAN tagging status.

	Command or Action	Purpose
	Example: Switch# <code>show vlan dot1q native</code>	
Step 11	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 Protocol Tunneling

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. Use one of the following:
 - `switchport mode access`
 - `switchport mode dot1q-tunnel`
5. `l2protocol-tunnel [cdp | lldp | point-to-point | stp | vtp]`
6. `l2protocol-tunnel shutdown-threshold [packet_second_rate_value | cdp | lldp point-to-point | stp | vtp]`
7. `l2protocol-tunnel drop-threshold [packet_second_rate_value | cdp | lldp | point-to-point | stp | vtp]`
8. `exit`
9. `errdisable recovery cause l2ptguard`
10. `l2protocol-tunnel cos value`
11. `end`
12. `show l2protocol`
13. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet1/0/1</code>	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode dot1q-tunnel Example: Switch# <code>switchport mode access</code> or Switch# <code>switchport mode dot1q-tunnel</code>	Configures the interface as an access port or an IEEE 802.1Q tunnel port.
Step 5	l2protocol-tunnel [cdp lldp point-to-point stp vtp] Example: Switch# <code>l2protocol-tunnel cdp</code>	Enables protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three Layer 2 protocols. Note Use the no l2protocol-tunnel [cdp lldp point-to-point stp vtp] interface configuration command to disable protocol tunneling for one of the Layer 2 protocols or for all three.
Step 6	l2protocol-tunnel shutdown-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp] Example: Switch# <code>l2protocol-tunnel shutdown-threshold 100 cdp</code>	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. Note If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.

	Command or Action	Purpose
		<p>Note Use the no l2protocol-tunnel shutdown-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp] and the no l2protocol-tunnel drop-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp] commands to return the shutdown and drop thresholds to the default settings.</p>
Step 7	<p>l2protocol-tunnel drop-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp]</p> <p>Example:</p> <pre>Switch# l2protocol-tunnel drop-threshold 100 cdp</pre>	<p>(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured.</p> <p>Note If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.</p> <p>Note Use the no l2protocol-tunnel shutdown-threshold [cdp lldp point-to-point stp vtp] and the no l2protocol-tunnel drop-threshold [cdp stp vtp] commands to return the shutdown and drop thresholds to the default settings.</p>
Step 8	<p>exit</p> <p>Example:</p> <pre>Switch# exit</pre>	Returns to global configuration mode.
Step 9	<p>errdisable recovery cause l2ptguard</p> <p>Example:</p> <pre>Switch(config)# errdisable recovery cause l2ptguard</pre>	(Optional) Configures the recovery mechanism from a Layer 2 maximum-rate error so that the interface is reenabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 10	<p>l2protocol-tunnel cos <i>value</i></p> <p>Example:</p> <pre>Switch(config)# l2protocol-tunnel cos value 7</pre>	(Optional) Configures the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.

	Command or Action	Purpose
Step 11	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 12	show l2protocol Example: <pre>Switch# show l2protocol</pre>	Displays the Layer 2 tunnel ports on the device, including the protocols configured, the thresholds, and the counters.
Step 13	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the SP Edge Switch

Before you begin

For EtherChannels, you need to configure both the SP (service-provider) edge devices and the customer devices for Layer 2 protocol tunneling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode dot1q-tunnel**
5. **l2protocol-tunnel point-to-point** [**pagp** | **lACP** | **udld**]
6. **l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lACP** | **udld**]] *value*
7. **l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** | **lACP** | **udld**]] *value*
8. **no cdp enable**
9. **spanning-tree bpdud filter enable**
10. **exit**
11. **errdisable recovery cause l2ptguard**
12. **l2protocol-tunnel cos** *value*
13. **end**
14. **show l2protocol**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/1</pre>	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	switchport mode dot1q-tunnel Example: <pre>Switch(config-if)# switchport mode dot1q-tunnel</pre>	Configures the interface as an IEEE 802.1Q tunnel port.
Step 5	l2protocol-tunnel point-to-point [pagp lacp udld] Example: <pre>Switch(config-if)# l2protocol-tunnel point-to-point pagp</pre>	(Optional) Enables point-to-point protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three protocols. <p>Note To avoid a network failure, make sure that the network is a point-to-point topology before you enable tunneling for PAGP, LACP, or UDLD packets.</p> <p>Note Use the no l2protocol-tunnel [point-to-point [pagp lacp udld]] interface configuration command to disable point-to-point protocol tunneling for one of the Layer 2 protocols or for all three.</p>
Step 6	l2protocol-tunnel shutdown-threshold [point-to-point [pagp lacp udld]] <i>value</i> Example: <pre>Switch(config-if)# l2protocol-tunnel shutdown-threshold point-to-point pagp 100</pre>	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured.

	Command or Action	Purpose
		<p>Note If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.</p> <p>Note Use the no l2protocol-tunnel shutdown-threshold [point-to-point [pagp lacp udld]] and the no l2protocol-tunnel drop-threshold [[point-to-point [pagp lacp udld]]] commands to return the shutdown and drop thresholds to the default settings.</p>
Step 7	<p>l2protocol-tunnel drop-threshold [point-to-point [pagp lacp udld]] value</p> <p>Example:</p> <pre>Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 500</pre>	<p>(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured.</p> <p>Note If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.</p>
Step 8	<p>no cdp enable</p> <p>Example:</p> <pre>Switch(config-if)# no cdp enable</pre>	Disables CDP on the interface.
Step 9	<p>spanning-tree bpdu filter enable</p> <p>Example:</p> <pre>Switch(config-if)# spanning-tree bpdu filter enable</pre>	Enables BPDU filtering on the interface.
Step 10	<p>exit</p> <p>Example:</p> <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 11	<p>errdisable recovery cause l2ptguard</p> <p>Example:</p> <pre>Switch(config)# errdisable recovery cause l2ptguard</pre>	(Optional) Configures the recovery mechanism from a Layer 2 maximum-rate error so that the interface is reenabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.

	Command or Action	Purpose
Step 12	l2protocol-tunnel cos <i>value</i> Example: <pre>Switch(config)# l2protocol-tunnel cos 2</pre>	(Optional) Configures the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 13	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 14	show l2protocol Example: <pre>Switch)# show l2protocol</pre>	Displays the Layer 2 tunnel ports on the device, including the protocols configured, the thresholds, and the counters.
Step 15	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the Customer Device

Before you begin

For EtherChannels, you need to configure both the SP edge device and the customer devices for Layer 2 protocol tunneling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport trunk encapsulation dot1q**
5. **switchport mode trunk**
6. **udld port**
7. **channel-group *channel-group-number* mode desirable**
8. **exit**
9. **interface port-channel *port-channel number***
10. **shutdown**
11. **no shutdown**
12. **end**
13. **show l2protocol**

14. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	switchport trunk encapsulation dot1q Example: Switch(config)# switchport trunk encapsulation dot1q	Sets the trunking encapsulation format to IEEE 802.1Q.
Step 5	switchport mode trunk Example: Switch(config-if)# switchport mode trunk	Enables trunking on the interface.
Step 6	udld port Example: Switch(config-if)# udld port	Enables UDLD in normal mode on the interface.
Step 7	channel-group <i>channel-group-number</i> mode desirable Example: Switch(config-if)# channel-group 25 mode desirable	Assigns the interface to a channel group, and specifies desirable for the PAgP mode.
Step 8	exit	Returns to global configuration mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(config-if)# exit</pre>	
Step 9	<p>interface port-channel port-channel <i>number</i></p> <p>Example:</p> <pre>Switch(config)# interface port-channel port-channel 25</pre>	Enters port-channel interface mode.
Step 10	<p>shutdown</p> <p>Example:</p> <pre>Switch(config)# shutdown</pre>	Shuts down the interface.
Step 11	<p>no shutdown</p> <p>Example:</p> <pre>Switch(config)# no shutdown</pre>	Enables the interface.
Step 12	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 13	<p>show l2protocol</p> <p>Example:</p> <pre>Switch# show l2protocol</pre>	Displays the Layer 2 tunnel ports on the device, including the protocols configured, the thresholds, and the counters.
Step 14	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>Note Use the no switchport mode trunk, the no udd enable, and the no channel group channel-group-number mode desirable interface configuration commands to return the interface to the default settings.</p>

Configuration Examples for IEEE 802.1Q and Layer 2 Protocol Tunneling

Example: Configuring an IEEE 802.1Q Tunneling Port

The following example shows how to configure an interface as a tunnel port, enable tagging of native VLAN packets, and verify the configuration.

```
Switch(config)# interface gigabitethernet1/0/7
Switch(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# exit
Switch(config)# vlan dot1q tag native
Switch(config)# end
Switch# show dot1q-tunnel interface gigabitethernet1/0/7
Port
-----
Gi1/0/1Port
-----
Switch# show vlan dot1q tag native
dot1q native vlan tagging is enabled
```

Example: Configuring Layer 2 Protocol Tunneling

The following example shows how to configure Layer 2 protocol tunneling for CDP, STP, and VTP and to verify the configuration.

```
Switch(config)# interface gigabitethernet1/0/11
Switch(config-if)# l2protocol-tunnel cdp
Switch(config-if)# l2protocol-tunnel stp
Switch(config-if)# l2protocol-tunnel vtp
Switch(config-if)# l2protocol-tunnel shutdown-threshold 1500
Switch(config-if)# l2protocol-tunnel drop-threshold 1000
Switch(config-if)# exit
Switch(config)# l2protocol-tunnel cos 7
Switch(config)# end
Switch# show l2protocol
```

```
COS for Encapsulated Packets: 7
Port Protocol Shutdown Drop Encapsulation Decapsulation Drop
Threshold Threshold Counter Counter Counter
-----
Gi0/11 cdp 1500 1000 2288 2282 0
stp 1500 1000 116 13 0
vtp 1500 1000 3 67 0
pagp ---- ---- 0 0 0
lacp ---- ---- 0 0 0
udld ---- ---- 0 0 0
```

Examples: Configuring the SP Edge and Customer Switches

This example shows how to configure the SP edge switch 1 and edge switch 2. VLANs 17, 18, 19, and 20 are the access VLANs, Fast Ethernet interfaces 1 and 2 are point-to-point tunnel ports with PAgP and UDLD enabled, the drop threshold is 1000, and Fast Ethernet interface 3 is a trunk port.

SP edge switch 1 configuration:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 17
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 18
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)#
Switch(config-if)# switchport mode trunk
```

SP edge switch 2 configuration:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 19
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 20
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)#
Switch(config-if)# switchport mode trunk
```

This example shows how to configure the customer switch at Site 1. Fast Ethernet interfaces 1, 2, 3, and 4 are set for IEEE 802.1Q trunking, UDLD is enabled, EtherChannel group 1 is enabled, and the port channel is shut down and then enabled to activate the EtherChannel configuration.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
```

```

Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# uddld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# uddld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# uddld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface port-channel 1
Switch(config-if)# shutdown
Switch(config-if)# no shutdown
Switch(config-if)# exit

```

Monitoring Tunneling Status

The following table describes the commands used to monitor tunneling status.

Table 40: Commands for Monitoring Tunneling

Command	Purpose
show dot1q-tunnel	Displays IEEE 802.1Q tunnel ports on the device.
show dot1q-tunnel interface <i>interface-id</i>	Verifies if a specific interface is a tunnel port.
show vlan dot1q tag native	Displays the status of native VLAN tagging on the device.

Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN Trunking
- Private VLANs
- VLAN Membership Policy Server (VMPS)
- Voice VLANs



CHAPTER 24

Configuring Spanning Tree Protocol

This chapter describes how to configure the Spanning Tree Protocol (STP) on port-based VLANs on the Catalyst devices. The device can use either the per-VLAN spanning-tree plus (PVST+) protocol based on the IEEE 802.1D standard and Cisco proprietary extensions, or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol based on the IEEE 802.1w standard. A switch stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID.

- [Finding Feature Information, on page 417](#)
- [Restrictions for STP, on page 417](#)
- [Information About Spanning Tree Protocol, on page 418](#)
- [How to Configure Spanning-Tree Features, on page 428](#)
- [Monitoring Spanning-Tree Status, on page 440](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restrictions for STP

- An attempt to configure a device as the root device fails if the value necessary to be the root device is less than 1.
- If your network consists of devices that support and do not support the extended system ID, it is unlikely that the device with the extended system ID support will become the root device. The extended system ID increases the device priority value every time the VLAN number is greater than the priority of the connected devices running older software.
- The root device for each spanning-tree instance should be a backbone or distribution device. Do not configure an access device as the spanning-tree primary root.

Information About Spanning Tree Protocol

Spanning Tree Protocol

Spanning Tree Protocol (STP) is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Devices might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

The STP uses a spanning-tree algorithm to select one device of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

The device that has *all* of its ports as the designated role or as the backup role is the root device. The device that has at least *one* of its ports in the designated role is called the designated device.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Devices send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The devices do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending device and its ports, including device and MAC addresses, device priority, port priority, and path cost. Spanning tree uses this information to elect the root device and root port for the switched network and the root port and designated port for each switched segment.

When two ports on a device are part of a loop, the spanning-tree and path cost settings control which port is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of a port in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.



Note By default, the device sends keepalive messages (to ensure the connection is up) only on interfaces that do not have small form-factor pluggable (SFP) modules. You can change the default for an interface by entering the `[no] keepalive` interface configuration command with no keywords.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is controlled by these elements:

- The unique bridge ID (device priority and MAC address) associated with each VLAN on each device.

- The spanning-tree path cost to the root device.
- The port identifier (port priority and MAC address) associated with each Layer 2 interface.

When the devices in a network are powered up, each functions as the root device. Each device sends a configuration BPDU through all of its ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the device that the sending device identifies as the root device
- The spanning-tree path cost to the root
- The bridge ID of the sending device
- Message age
- The identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a device receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the device, the device also forwards it with an updated message to all attached LANs for which it is the designated device.

If a device receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the device is a designated device for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One device in the network is elected as the root device (the logical center of the spanning-tree topology in a switched network). See the figure following the bullets.
For each VLAN, the device with the highest device priority (the lowest numerical priority value) is elected as the root device. If all devices are configured with the default priority (32768), the device with the lowest MAC address in the VLAN becomes the root device. The device priority value occupies the most significant bits of the bridge ID, .
- A root port is selected for each device (except the root device). This port provides the best path (lowest cost) when the device forwards packets to the root device.
- The shortest distance to the root device is calculated for each device based on the path cost.
- A designated device for each LAN segment is selected. The designated device incurs the lowest path cost when forwarding packets from that LAN to the root device. The port through which the designated device is attached to the LAN is called the designated port.



Note If the **logging event spanning tree** command is configured on multiple interfaces and the topology changes, it may result in several logging messages and high CPU utilization. This may cause the switch to drop or delay the processing of STP BPDUs.

To prevent this behavior, remove the **logging event spanning tree** and **logging event status** commands or disable logging to the console.

All paths that are not needed to reach the root device from anywhere in the switched network are placed in the spanning-tree blocking mode.

Bridge ID, Device Priority, and Extended System ID

The IEEE 802.1D standard requires that each device has a unique bridge identifier (bridge ID), which controls the selection of the root device. Because each VLAN is considered as a different *logical bridge* with PVST+ and Rapid PVST+, the same device must have a different bridge ID for each configured VLAN. Each VLAN on the device has a unique 8-byte bridge ID. The 2 most-significant bytes are used for the device priority, and the remaining 6 bytes are derived from the device MAC address.

The device supports the IEEE 802.1t spanning-tree extensions, and some of the bits previously used for the device priority are now used as the VLAN identifier. The result is that fewer MAC addresses are reserved for the device, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID.

The 2 bytes previously used for the device priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the VLAN ID.

Table 41: Device Priority Value and Extended System ID

Priority Value				Extended System ID (Set Equal to the VLAN ID)										
Bit 16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2

Spanning tree uses the extended system ID, the device priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN.

Support for the extended system ID affects how you manually configure the root device, the secondary root device, and the device priority of a VLAN. For example, when you change the device priority value, you change the probability that the device will be elected as the root device. Configuring a higher value decreases the probability; a lower value increases the probability.

If any root device for the specified VLAN has a device priority lower than 24576, the device sets its own priority for the specified VLAN to 4096 less than the lowest device priority. 4096 is the value of the least-significant bit of a 4-bit device priority value as shown in the table.

Port Priority Versus Path Cost

If a loop occurs, spanning tree uses port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

The spanning-tree path cost default value is derived from the media speed of an interface. If a loop occurs, spanning tree uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

If your device is a member of a device stack, you must assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last instead of adjusting its port priority. For details, see Related Topics.

Spanning-Tree Interface States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When an interface transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

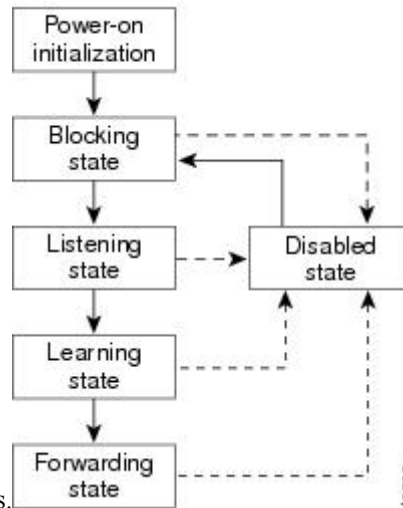
Each Layer 2 interface on a device using spanning tree exists in one of these states:

- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree decides that the interface should participate in frame forwarding.
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

An interface moves through these states:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

Figure 21: Spanning-Tree Interface States



An interface moves through the states.

When you power up the device, spanning tree is enabled by default, and every interface in the device, VLAN, or network goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

When the spanning-tree algorithm places a Layer 2 interface in the forwarding state, this process occurs:

1. The interface is in the listening state while spanning tree waits for protocol information to move the interface to the blocking state.
2. While spanning tree waits for the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
3. In the learning state, the interface continues to block frame forwarding as the device learns end-station location information for the forwarding database.
4. When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each device interface. A device initially functions as the root until it exchanges BPDUs with other devices. This exchange establishes which device in the network is the root or root device. If there is only one device in the network, no exchange occurs, the forward-delay timer expires, and the interface moves to the listening state. An interface always enters the blocking state after device initialization.

An interface in the blocking state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree decides that the interface should participate in frame forwarding.

An interface in the listening state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

A Layer 2 interface in the forwarding state forwards frames. The interface enters the forwarding state from the learning state.

An interface in the forwarding state performs these functions:

- Receives and forwards frames received on the interface
- Forwards frames switched from another interface
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

A disabled interface performs these functions:

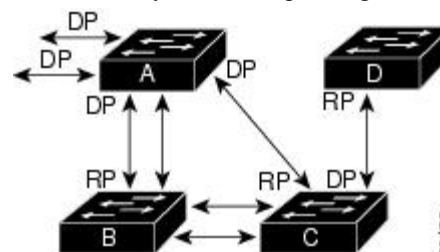
- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Does not receive BPDUs

How a Device or Port Becomes the Root Device or Root Port

If all devices in a network are enabled with default spanning-tree settings, the device with the lowest MAC address becomes the root device.

Figure 22: Spanning-Tree Topology

Device A is elected as the root device because the device priority of all the devices is set to the default (32768) and Device A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Device A might not be the ideal root device. By increasing the priority (lowering the numerical value) of the ideal device so that it becomes the root device, you force a spanning-tree recalculation



to form a new topology with the ideal device as the root. DP = Designated Port

When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the root port can cause a root-port change. The goal is to make the fastest link the root port.

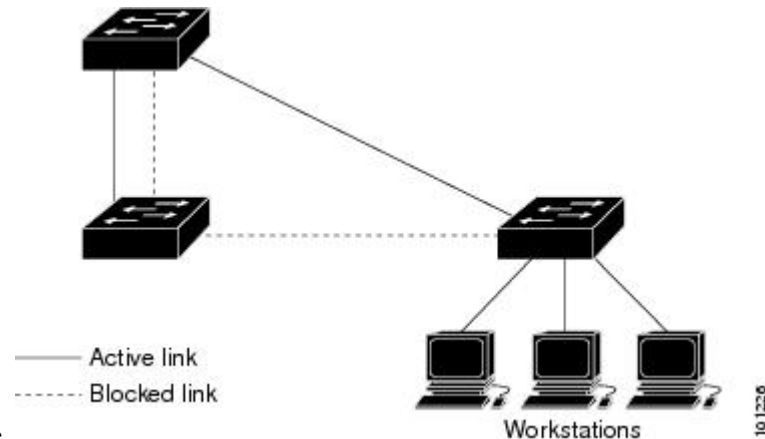
For example, assume that one port on Device B is a Gigabit Ethernet link and that another port on Device B (a 10/100 link) is the root port. Network traffic might be more efficient over the Gigabit Ethernet link. By changing the spanning-tree port priority on the Gigabit Ethernet port to a higher priority (lower numerical value) than the root port, the Gigabit Ethernet port becomes the new root port.

Spanning Tree and Redundant Connectivity

Figure 23: Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two device interfaces to another device or to two different devices. Spanning tree automatically disables one interface but enables it if the other one fails. If one link is high-speed and the other is low-speed, the low-speed link is always disabled. If the speeds

are the same, the port priority and port ID are added together, and spanning tree disables the link with the



highest value.

You can also create redundant links between devices by using EtherChannel groups.

Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C2000000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

Regardless of the spanning-tree state, each device in the stack receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the CPU on the device or on each device in the stack receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning tree is disabled, the device or each device in the stack forwards those packets as unknown multicast addresses.

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, the default setting of the **mac address-table aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned. The accelerated aging is the same as the forward-delay parameter value (**spanning-tree vlan *vlan-id* forward-time *seconds*** global configuration command) when the spanning tree reconfigures.

Because each VLAN is a separate spanning-tree instance, the device accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the device.

Spanning-Tree Modes and Protocols

The device supports these spanning-tree modes and protocols:

- PVST+—This spanning-tree mode is based on the IEEE 802.1D standard and Cisco proprietary extensions. The PVST+ runs on each VLAN on the device up to the maximum supported, ensuring that each has a loop-free path through the network.

The PVST+ provides Layer 2 load-balancing for the VLAN on which it runs. You can create different logical topologies by using the VLANs on your network to ensure that all of your links are used but that no one link is oversubscribed. Each instance of PVST+ on a VLAN has a single root device. This root

device propagates the spanning-tree information associated with that VLAN to all other devices in the network. Because each device has the same information about the network, this process ensures that the network topology is maintained.

- **Rapid PVST+**—This spanning-tree mode is the same as PVST+ except that it uses a rapid convergence based on the IEEE 802.1w standard. Beginning from 15.2(4)E release, the STP default mode is Rapid PVST+. To provide rapid convergence, the Rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

Rapid PVST+ uses the same configuration as PVST+ (except where noted), and the device needs only minimal extra configuration. The benefit of Rapid PVST+ is that you can migrate a large PVST+ install base to Rapid PVST+ without having to learn the complexities of the Multiple Spanning Tree Protocol (MSTP) configuration and without having to re-provision your network. In Rapid PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

- **MSTP**—This spanning-tree mode is based on the IEEE 802.1s standard. You can map multiple VLANs to the same spanning-tree instance, which reduces the number of spanning-tree instances required to support a large number of VLANs. The MSTP runs on top of the RSTP (based on IEEE 802.1w), which provides for rapid convergence of the spanning tree by eliminating the forward delay and by quickly transitioning root ports and designated ports to the forwarding state. In a device stack, the cross-stack rapid transition (CSRT) feature performs the same function as RSTP. You cannot run MSTP without RSTP or CSRT.

Supported Spanning-Tree Instances

In PVST+ or Rapid PVST+ mode, the device or device stack supports up to 128 spanning-tree instances.

In MSTP mode, the device or device stack supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.

Spanning-Tree Interoperability and Backward Compatibility

In a mixed MSTP and PVST+ network, the common spanning-tree (CST) root must be inside the MST backbone, and a PVST+ device cannot connect to multiple MST regions.

When a network contains devices running Rapid PVST+ and devices running PVST+, we recommend that the Rapid PVST+ devices and PVST+ devices be configured for different spanning-tree instances. In the Rapid PVST+ spanning-tree instances, the root device must be a Rapid PVST+ device. In the PVST+ instances, the root device must be a PVST+ device. The PVST+ devices should be at the edge of the network.

All stack members run the same version of spanning tree (all PVST+, all Rapid PVST+, or all MSTP).

Table 42: PVST+, MSTP, and Rapid-PVST+ Interoperability and Compatibility

	PVST+	MSTP	Rapid PVST+
PVST+	Yes	Yes (with restrictions)	Yes (reverts to PVST+)
MSTP	Yes (with restrictions)	Yes	Yes (reverts to PVST+)
Rapid PVST+	Yes (reverts to PVST+)	Yes (reverts to PVST+)	Yes

STP and IEEE 802.1Q Trunks

The IEEE 802.1Q standard for VLAN trunks imposes some limitations on the spanning-tree strategy for a network. The standard requires only one spanning-tree instance for *all* VLANs allowed on the trunks. However, in a network of Cisco devices connected through IEEE 802.1Q trunks, the devices maintain one spanning-tree instance for *each* VLAN allowed on the trunks.

When you connect a Cisco device to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco device uses PVST+ to provide spanning-tree interoperability. If Rapid PVST+ is enabled, the device uses it instead of PVST+. The device combines the spanning-tree instance of the IEEE 802.1Q VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q device.

However, all PVST+ or Rapid PVST+ information is maintained by Cisco devices separated by a cloud of non-Cisco IEEE 802.1Q devices. The non-Cisco IEEE 802.1Q cloud separating the Cisco devices is treated as a single trunk link between the devices.

Rapid PVST+ is automatically enabled on IEEE 802.1Q trunks, and no user configuration is required. The external spanning-tree behavior on access ports and Inter-Switch Link (ISL) trunk ports is not affected by PVST+.

VLAN-Bridge Spanning Tree

Cisco VLAN-bridge spanning tree is used with the fallback bridging feature (bridge groups), which forwards non-IP protocols such as DECnet between two or more VLAN bridge domains or routed ports. The VLAN-bridge spanning tree allows the bridge groups to form a spanning tree on top of the individual VLAN spanning trees to prevent loops from forming if there are multiple connections among VLANs. It also prevents the individual spanning trees from the VLANs being bridged from collapsing into a single spanning tree.

To support VLAN-bridge spanning tree, some of the spanning-tree timers are increased. To use the fallback bridging feature, you must have the IP services feature set enabled on your device.

Default Spanning-Tree Configuration

Table 43: Default Spanning-Tree Configuration

Feature	Default Setting
Enable state	Enabled on VLAN 1.
Spanning-tree mode	Rapid PVST+ (PVST+ and MST disabled.)
Device priority	32768
Spanning-tree port priority (configurable on a per-interface basis)	128
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mb/s: 4 100 Mb/s: 19 10 Mb/s: 100
Spanning-tree VLAN port priority (configurable on a per-VLAN basis)	128

Feature	Default Setting
Spanning-tree VLAN port cost (configurable on a per-VLAN basis)	1000 Mb/s: 4 100 Mb/s: 19 10 Mb/s: 100
Spanning-tree timers	Hello time: 2 seconds Forward-delay time: 15 seconds Maximum-aging time: 20 seconds Transmit hold count: 6 BPDUs



Note Beginning in Cisco IOS Release 15.2(4)E, the default STP mode is Rapid PVST+.

How to Configure Spanning-Tree Features

Changing the Spanning-Tree Mode

The switch supports three spanning-tree modes: per-VLAN spanning tree plus (PVST+), Rapid PVST+, or multiple spanning tree protocol (MSTP). By default, the device runs the Rapid PVST+ protocol.

If you want to enable a mode that is different from the default mode, this procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mode {pvst | mst | rapid-pvst}**
4. **interface *interface-id***
5. **spanning-tree link-type point-to-point**
6. **end**
7. **clear spanning-tree detected-protocols**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree mode {pvst mst rapid-pvst} Example: <pre>Switch(config)# spanning-tree mode pvst</pre>	Configures a spanning-tree mode. All stack members run the same version of spanning tree. <ul style="list-style-type: none"> • Select pvst to enable PVST+. • Select mst to enable MSTP. • Select rapid-pvst to enable rapid PVST+.
Step 4	interface interface-id Example: <pre>Switch(config)# interface GigabitEthernet1/0/1</pre>	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port channels. The VLAN ID range is 1 to 4094. The port-channel range is 1 to .
Step 5	spanning-tree link-type point-to-point Example: <pre>Switch(config-if)# spanning-tree link-type point-to-point</pre>	Specifies that the link type for this port is point-to-point. If you connect this port (local port) to a remote port through a point-to-point link and the local port becomes a designated port, the device negotiates with the remote port and rapidly changes the local port to the forwarding state.
Step 6	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 7	clear spanning-tree detected-protocols Example: <pre>Switch# clear spanning-tree detected-protocols</pre>	If any port on the device is connected to a port on a legacy IEEE 802.1D device, this command restarts the protocol migration process on the entire device. This step is optional if the designated device detects that this device is running rapid PVST+.

Disabling Spanning Tree

Spanning tree is enabled by default on VLAN 1 and on all newly created VLANs up to the spanning-tree limit. Disable spanning tree only if you are sure there are no loops in the network topology.



Caution When spanning tree is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree vlan *vlan-id***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	no spanning-tree vlan <i>vlan-id</i> Example: Switch(config)# no spanning-tree vlan 300	For <i>vlan-id</i> , the range is 1 to 4094.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Configuring the Root Device

To configure a device as the root for the specified VLAN, use the **spanning-tree vlan *vlan-id* root** global configuration command to modify the device priority from the default value (32768) to a significantly lower value. When you enter this command, the software checks the device priority of the root devices for each VLAN. Because of the extended system ID support, the device sets its own priority for the specified VLAN to 24576 if this value will cause this device to become the root for the specified VLAN.

Use the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of device hops between any two end stations in the Layer 2 network). When you specify the network diameter, the device automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `spanning-tree vlan vlan-id root primary [diameter net-diameter]`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>spanning-tree vlan <i>vlan-id</i> root primary [<i>diameter net-diameter</i>]</code></p> <p>Example:</p> <pre>Switch(config)# spanning-tree vlan 20-24 root primary diameter 4</pre>	<p>Configures a device to become the root for the specified VLAN.</p> <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For <i>diameter net-diameter</i>, specify the maximum number of devices between any two end stations. The range is 2 to 7.
Step 4	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

What to do next

After configuring the device as the root device, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **`spanning-tree vlan vlan-id hello-time`**, **`spanning-tree vlan vlan-id forward-time`**, and the **`spanning-tree vlan vlan-id max-age`** global configuration commands.

Configuring a Secondary Root Device

When you configure a device as the secondary root, the device priority is modified from the default value (32768) to 28672. With this priority, the device is likely to become the root device for the specified VLAN if the primary root device fails. This is assuming that the other network devices use the default device priority of 32768, and therefore, are unlikely to become the root device.

You can execute this command on more than one device to configure multiple backup root devices. Use the same network diameter and hello-time values that you used when you configured the primary root device with the **spanning-tree vlan *vlan-id* root primary** global configuration command.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* root secondary [diameter *net-diameter*]**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> root secondary [diameter <i>net-diameter</i>] Example: <pre>Switch(config)# spanning-tree vlan 20-24 root secondary diameter 4</pre>	Configures a device to become the secondary root for the specified VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For diameter <i>net-diameter</i>, specify the maximum number of devices between any two end stations. The range is 2 to 7. Use the same network diameter value that you used when configuring the primary root device.
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	

Configuring Port Priority



Note If your device is a member of a device stack, you must use the **spanning-tree [vlan *vlan-id*] cost *cost*** interface configuration command instead of the **spanning-tree [vlan *vlan-id*] port-priority *priority*** interface configuration command to select an interface to put in the forwarding state. Assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **spanning-tree port-priority *priority***
5. **spanning-tree vlan *vlan-id* port-priority *priority***
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (port-channel <i>port-channel-number</i>).
Step 4	spanning-tree port-priority <i>priority</i> Example:	Configures the port priority for an interface. For <i>priority</i> , the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96,

	Command or Action	Purpose
	Switch(config-if) # spanning-tree port-priority 0	112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 5	spanning-tree vlan <i>vlan-id</i> port-priority <i>priority</i> Example: Switch(config-if) # spanning-tree vlan 20-25 port-priority 0	Configures the port priority for a VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>priority</i>, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 6	end Example: Switch(config-if) # end	Returns to privileged EXEC mode.

Configuring Path Cost

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **spanning-tree cost *cost***
5. **spanning-tree vlan *vlan-id* cost *cost***
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	<code>interface interface-id</code> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (port-channel port-channel-number).
Step 4	<code>spanning-tree cost cost</code> Example: Switch(config-if)# <code>spanning-tree cost 250</code>	Configures the cost for an interface. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. For <i>cost</i> , the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	<code>spanning-tree vlan vlan-id cost cost</code> Example: Switch(config-if)# <code>spanning-tree vlan 10,12-15,20 cost 300</code>	Configures the cost for a VLAN. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. <ul style="list-style-type: none"> For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. For <i>cost</i>, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 6	<code>end</code> Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.

The `show spanning-tree interface interface-id` privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the `show running-config` privileged EXEC command to confirm the configuration.

Configuring the Device Priority of a VLAN

You can configure the device priority and make it more likely that a standalone device or a device in the stack will be chosen as the root device.



Note Exercise care when using this command. For most situations, we recommend that you use the `spanning-tree vlan vlan-id root primary` and the `spanning-tree vlan vlan-id root secondary` global configuration commands to modify the device priority.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* priority *priority***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> priority <i>priority</i> Example: Switch(config)# spanning-tree vlan 20 priority 8192	Configures the device priority of a VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>priority</i>, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the device will be chosen as the root device. Valid priority values are 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 4	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root device.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **spanning-tree vlan** *vlan-id* **hello-time** *seconds*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	spanning-tree vlan <i>vlan-id</i> hello-time <i>seconds</i> Example: Switch(config)# spanning-tree vlan 20-24 hello-time 3	Configures the hello time of a VLAN. The hello time is the time interval between configuration messages generated and sent by the root device. These messages mean that the device is alive. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>seconds</i>, the range is 1 to 10; the default is 2.
Step 3	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Configuring the Forwarding-Delay Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **forward-time** *seconds*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> forward-time <i>seconds</i> Example: Switch(config)# spanning-tree vlan 20,25 forward-time 18	Configures the forward time of a VLAN. The forwarding delay is the number of seconds an interface waits before changing from its spanning-tree learning and listening states to the forwarding state. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>seconds</i>, the range is 4 to 30; the default is 15.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Configuring the Maximum-Aging Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* max-age *seconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> max-age <i>seconds</i> Example: <pre>Switch(config)# spanning-tree vlan 20 max-age 30</pre>	Configures the maximum-aging time of a VLAN. The maximum-aging time is the number of seconds a device waits without receiving spanning-tree configuration messages before attempting a reconfiguration. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>seconds</i>, the range is 6 to 40; the default is 20.
Step 4	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring the Transmit Hold-Count

You can configure the BPDU burst size by changing the transmit hold count value.



Note Changing this parameter to a higher value can have a significant impact on CPU utilization, especially in Rapid PVST+ mode. Lowering this value can slow down convergence in certain scenarios. We recommend that you maintain the default setting.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree transmit hold-count *value***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree transmit hold-count <i>value</i> Example: Switch(config)# spanning-tree transmit hold-count 6	Configures the number of BPDUs that can be sent before pausing for 1 second. For <i>value</i> , the range is 1 to 20; the default is 6.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Monitoring Spanning-Tree Status

Table 44: Commands for Displaying Spanning-Tree Status

show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree vlan <i>vlan-id</i>	Displays spanning-tree information for the specified VLAN.
show spanning-tree interface <i>interface-id</i>	Displays spanning-tree information for the specified interface.
show spanning-tree interface <i>interface-id</i> portfast	Displays spanning-tree portfast information for the specified interface.
show spanning-tree summary [totals]	Displays a summary of interface states or displays the total lines of the state section.

To clear spanning-tree counters, use the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.



CHAPTER 25

Configuring Multiple Spanning-Tree Protocol

- [Finding Feature Information, on page 441](#)
- [Prerequisites for MSTP, on page 441](#)
- [Restrictions for MSTP, on page 442](#)
- [Information About MSTP, on page 442](#)
- [How to Configure MSTP Features, on page 458](#)
- [Examples, on page 476](#)
- [Monitoring MST Configuration and Status, on page 480](#)
- [Feature Information for MSTP, on page 480](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for MSTP

- For two or more devices to be in the same multiple spanning tree (MST) region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- For load-balancing across redundant paths in the network to work, all VLAN-to-instance mapping assignments must match; otherwise, all traffic flows on a single link.
- For load-balancing between a per-VLAN spanning tree plus (PVST+) and an MST cloud or between a rapid-PVST+ and an MST cloud to work, all MST boundary ports must be forwarding. MST boundary ports are forwarding when the root of the internal spanning tree (IST) of the MST cloud is the root of the common spanning tree (CST). If the MST cloud consists of multiple MST regions, one of the MST regions must contain the CST root, and all of the other MST regions must have a better path to the root contained within the MST cloud than a path through the PVST+ or rapid-PVST+ cloud. You might have to manually configure the devices in the clouds.

Restrictions for MSTP

- The device stack supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.
- PVST+, Rapid PVST+, and MSTP are supported, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run Rapid PVST+, or all VLANs run MSTP.)
- VLAN Trunking Protocol (VTP) propagation of the MST configuration is not supported. However, you can manually configure the MST configuration (region name, revision number, and VLAN-to-instance mapping) on each device within the MST region by using the command-line interface (CLI) or through the Simple Network Management Protocol (SNMP) support.
- Partitioning the network into a large number of regions is not recommended. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by routers or non-Layer 2 devices.
- A region can have one member or multiple members with the same MST configuration; each member must be capable of processing rapid spanning tree protocol (RSTP) Bridge Protocol Data Units (BPDUs). There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

Information About MSTP

MSTP Configuration

MSTP, which uses RSTP for rapid convergence, enables multiple VLANs to be grouped into and mapped to the same spanning-tree instance, reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths).



Note The multiple spanning-tree (MST) implementation is based on the IEEE 802.1s standard.

The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. This deployment provides the highly available network required in a service-provider environment.

When the device is in the MST mode, the RSTP, which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Both MSTP and RSTP improve the spanning-tree operation and maintain backward compatibility with equipment that is based on the (original) IEEE 802.1D spanning tree, with existing Cisco-proprietary Multiple Instance STP (MISTP), and with existing Cisco PVST+ and rapid per-VLAN spanning-tree plus (Rapid PVST+).

A device stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same device ID.

MSTP Configuration Guidelines

- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled.
- For configuration guidelines about UplinkFast, BackboneFast, and cross-stack UplinkFast, see the relevant sections in the Related Topics section.
- When the device is in MST mode, it uses the long path-cost calculation method (32 bits) to compute the path cost values. With the long path-cost calculation method, the following path cost values are supported:

Speed	Path Cost Value
10 Mb/s	2,000,000
100 Mb/s	200,000
1 Gb/s	20,000
10 Gb/s	2,000
100 Gb/s	200

Root Switch

The device maintains a spanning-tree instance for the group of VLANs mapped to it. A device ID, consisting of the device priority and the device MAC address, is associated with each instance. For a group of VLANs, the device with the lowest device ID becomes the root device.

When you configure a device as the root, you modify the device priority from the default value (32768) to a significantly lower value so that the device becomes the root device for the specified spanning-tree instance. When you enter this command, the device checks the device priorities of the root devices. Because of the extended system ID support, the device sets its own priority for the specified instance to 24576 if this value will cause this devices to become the root for the specified spanning-tree instance.

If any root device for the specified instance has a device priority lower than 24576, the device sets its own priority to 4096 less than the lowest device priority. (4096 is the value of the least-significant bit of a 4-bit device priority value. For more information, select "Bridge ID, Device Priority, and Extended System ID" link in Related Topics.

If your network consists of devices that support and do not support the extended system ID, it is unlikely that the device with the extended system ID support will become the root device. The extended system ID increases the device priority value every time the VLAN number is greater than the priority of the connected switches running older software.

The root device for each spanning-tree instance should be a backbone or distribution device. Do not configure an access device as the spanning-tree primary root.

Use the **diameter** keyword, which is available only for MST instance 0, to specify the Layer 2 network diameter (that is, the maximum number of device hops between any two end stations in the Layer 2 network). When you specify the network diameter, the device automatically sets an optimal hello time, forward-delay

time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Multiple Spanning-Tree Regions

For switches to participate in multiple spanning-tree (MST) instances, you must consistently configure the switches with the same MST configuration information. A collection of interconnected switches that have the same MST configuration comprises an MST region.

The MST configuration controls to which MST region each device belongs. The configuration includes the name of the region, the revision number, and the MST VLAN-to-instance assignment map. You configure the device for a region by specifying the MST region configuration on it. You can map VLANs to an MST instance, specify the region name, and set the revision number. For instructions and an example, select the "Specifying the MST Region Configuration and Enabling MSTP" link in Related Topics.

A region can have one or multiple members with the same MST configuration. Each member must be capable of processing RSTP bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network, but each region can support up to 65 spanning-tree instances. Instances can be identified by any number in the range from 0 to 4094. You can assign a VLAN to only one spanning-tree instance at a time.

IST, CIST, and CST

Unlike PVST+ and Rapid PVST+ in which all the spanning-tree instances are independent, the MSTP establishes and maintains two types of spanning trees:

- An internal spanning tree (IST), which is the spanning tree that runs in an MST region.

Within each MST region, the MSTP maintains multiple spanning-tree instances. Instance 0 is a special instance for a region, known as the internal spanning tree (IST). All other MST instances are numbered from 1 to 4094.

The IST is the only spanning-tree instance that sends and receives BPDUs. All of the other spanning-tree instance information is contained in M-records, which are encapsulated within MSTP BPDUs. Because the MSTP BPDUs carry information for all instances, the number of BPDUs that need to be processed to support multiple spanning-tree instances is significantly reduced.

All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root device ID, root path cost, and so forth. By default, all VLANs are assigned to the IST.

An MST instance is local to the region; for example, MST instance 1 in region A is independent of MST instance 1 in region B, even if regions A and B are interconnected.

- A common and internal spanning tree (CIST), which is a collection of the ISTs in each MST region, and the common spanning tree (CST) that interconnects the MST regions and single spanning trees.

The spanning tree computed in a region appears as a subtree in the CST that encompasses the entire switched domain. The CIST is formed by the spanning-tree algorithm running among switches that support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D standards. The CIST inside an MST region is the same as the CST outside a region.

Operations Within an MST Region

The IST connects all the MSTP switches in a region. When the IST converges, the root of the IST becomes the CIST regional root. It is the device within the region with the lowest device ID and path cost to the CIST root. The CIST regional root is also the CIST root if there is only one region in the network. If the CIST root is outside the region, one of the MSTP switches at the boundary of the region is selected as the CIST regional root.

When an MSTP device initializes, it sends BPDUs claiming itself as the root of the CIST and the CIST regional root, with both of the path costs to the CIST root and to the CIST regional root set to zero. The device also initializes all of its MST instances and claims to be the root for all of them. If the device receives superior MST root information (lower device ID, lower path cost, and so forth) than currently stored for the port, it relinquishes its claim as the CIST regional root.

During initialization, a region might have many subregions, each with its own CIST regional root. As switches receive superior IST information, they leave their old subregions and join the new subregion that contains the true CIST regional root. All subregions shrink except for the one that contains the true CIST regional root.

For correct operation, all switches in the MST region must agree on the same CIST regional root. Therefore, any two switches in the region only synchronize their port roles for an MST instance if they converge to a common CIST regional root.

Operations Between MST Regions

If there are multiple regions or legacy IEEE 802.1D devices within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP devices in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all the MSTP devices in the region and appears as a subtree in the CIST that encompasses the entire switched domain. The root of the subtree is the CIST regional root. The MST region appears as a virtual device to adjacent STP devices and MST regions.

Only the CST instance sends and receives BPDUs, and MST instances add their spanning-tree information into the BPDUs to interact with neighboring devices and compute the final spanning-tree topology. Because of this, the spanning-tree parameters related to BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MST instances. Parameters related to the spanning-tree topology (for example, device priority, port VLAN cost, and port VLAN priority) can be configured on both the CST instance and the MST instance.

MSTP devices use Version 3 RSTP BPDUs or IEEE 802.1D STP BPDUs to communicate with legacy IEEE 802.1D devices. MSTP devices use MSTP BPDUs to communicate with MSTP devices.

IEEE 802.1s Terminology

Some MST naming conventions used in Cisco's prestandard implementation have been changed to identify some *internal* or *regional* parameters. These parameters are significant only within an MST region, as opposed to external parameters that are relevant to the whole network. Because the CIST is the only spanning-tree instance that spans the whole network, only the CIST parameters require the external rather than the internal or regional qualifiers.

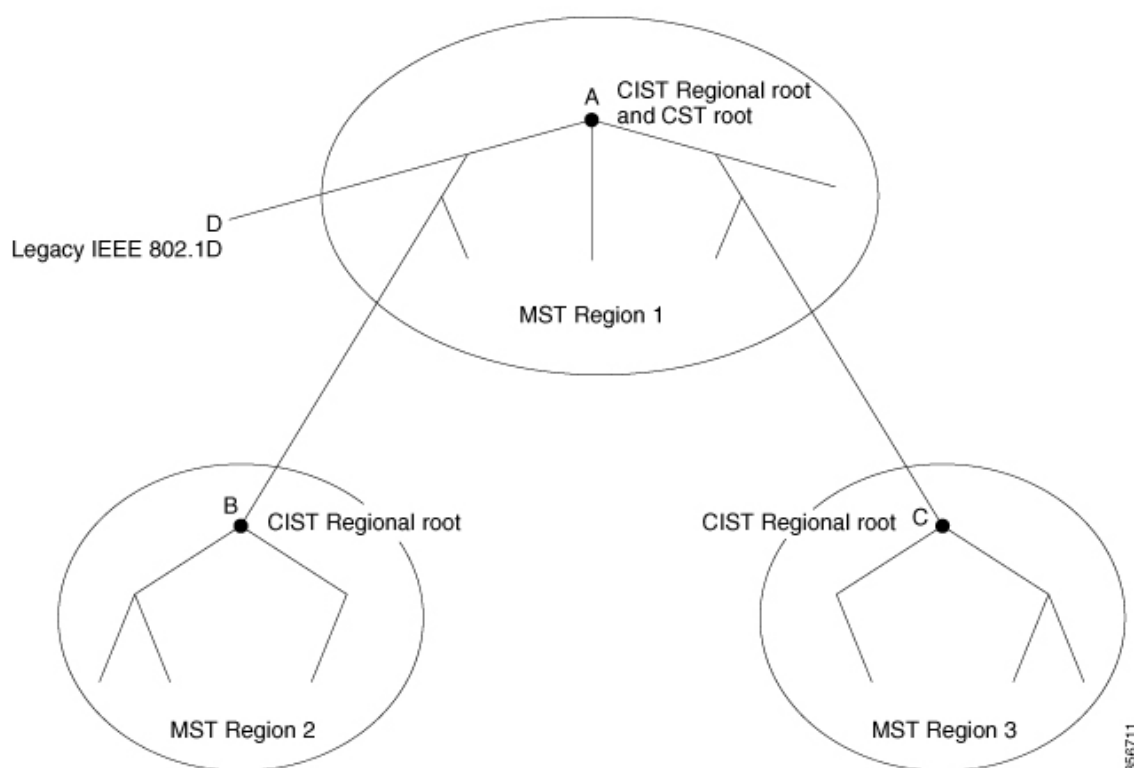
- The CIST root is the root device for the unique instance that spans the whole network, the CIST.
- The CIST external root path cost is the cost to the CIST root. This cost is left unchanged within an MST region. Remember that an MST region looks like a single device for the CIST. The CIST external root path cost is the root path cost calculated between these virtual devices and devices that do not belong to any region.

- If the CIST root is in the region, the CIST regional root is the CIST root. Otherwise, the CIST regional root is the closest device to the CIST root in the region. The CIST regional root acts as a root device for the IST.
- The CIST internal root path cost is the cost to the CIST regional root in a region. This cost is only relevant to the IST, instance 0.

Illustration of MST Regions

This figure displays three MST regions and a legacy IEEE 802.1D device (D). The CIST regional root for region 1 (A) is also the CIST root. The CIST regional root for region 2 (B) and the CIST regional root for region 3 (C) are the roots for their respective subtrees within the CIST. The RSTP runs in all regions.

Figure 24: MST Regions, CIST Regional Root, and CIST Root



Hop Count

The IST and MST instances do not use the message-age and maximum-age information in the configuration BPDU to compute the spanning-tree topology. Instead, they use the path cost to the root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism.

By using the **spanning-tree mst max-hops** global configuration command, you can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region. The hop count achieves the same result as the message-age information (triggers a reconfiguration). The root device of the instance always sends a BPDU (or M-record) with a cost of 0 and the hop count set to the maximum value. When a

device receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs it generates. When the count reaches zero, the device discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the RSTP portion of the BPDU remain the same throughout the region, and the same values are propagated by the region designated ports at the boundary.

Boundary Ports

In the Cisco prestandard implementation, a boundary port connects an MST region to a single spanning-tree region running RSTP, to a single spanning-tree region running PVST+ or rapid PVST+, or to another MST region with a different MST configuration. A boundary port also connects to a LAN, the designated device of which is either a single spanning-tree device or a device with a different MST configuration.

There is no definition of a boundary port in the IEEE 802.1s standard. The IEEE 802.1Q-2002 standard identifies two kinds of messages that a port can receive:

- internal (coming from the same region)
- external (coming from another region)

When a message is internal, the CIST part is received by the CIST, and each MST instance receives its respective M-record.

When a message is external, it is received only by the CIST. If the CIST role is root or alternate, or if the external BPDU is a topology change, it could have an impact on the MST instances.

An MST region includes both devices and LANs. A segment belongs to the region of its designated port. Therefore, a port in a different region than the designated port for a segment is a boundary port. This definition allows two ports internal to a region to share a segment with a port belonging to a different region, creating the possibility of a port receiving both internal and external messages.

The primary change from the Cisco prestandard implementation is that a designated port is not defined as boundary, unless it is running in an STP-compatible mode.



Note If there is a legacy STP device on the segment, messages are always considered external.

The other change from the Cisco prestandard implementation is that the CIST regional root device ID field is now inserted where an RSTP or legacy IEEE 802.1Q device has the sender device ID. The whole region performs like a single virtual device by sending a consistent sender device ID to neighboring devices. In this example, device C would receive a BPDU with the same consistent sender device ID of root, whether or not A or B is designated for the segment.

IEEE 802.1s Implementation

The Cisco implementation of the IEEE MST standard includes features required to meet the standard, as well as some of the desirable prestandard functionality that is not yet incorporated into the published standard.

Port Role Naming Change

The boundary role is no longer in the final MST standard, but this boundary concept is maintained in Cisco's implementation. However, an MST instance port at a boundary of the region might not follow the state of the corresponding CIST port. Two boundary roles currently exist:

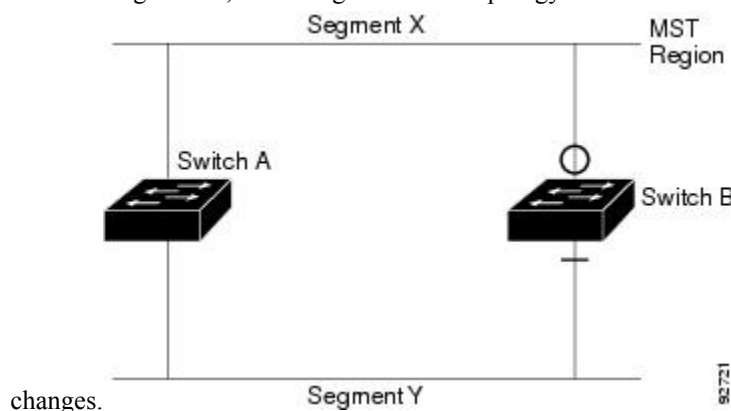
- The boundary port is the root port of the CIST regional root—When the CIST instance port is proposed and is in sync, it can send back an agreement and move to the forwarding state only after all the corresponding MSTI ports are in sync (and thus forwarding). The MSTI ports now have a special *primary* role.
- The boundary port is not the root port of the CIST regional root—The MSTI ports follow the state and role of the CIST port. The standard provides less information, and it might be difficult to understand why an MSTI port can be alternately blocking when it receives no BPDUs (MRecords). In this case, although the boundary role no longer exists, the **show** commands identify a port as boundary in the *type* column of the output.

Interoperation Between Legacy and Standard Devices

Because automatic detection of prestandard devices can fail, you can use an interface configuration command to identify prestandard ports. A region cannot be formed between a standard and a prestandard device, but they can interoperate by using the CIST. Only the capability of load-balancing over different instances is lost in that particular case. The CLI displays different flags depending on the port configuration when a port receives prestandard BPDUs. A syslog message also appears the first time a device receives a prestandard BPDU on a port that has not been configured for prestandard BPDU transmission.

Figure 25: Standard and Prestandard Device Interoperation

Assume that A is a standard device and B a prestandard device, both configured to be in the same region. A is the root device for the CIST, and B has a root port (BX) on segment X and an alternate port (BY) on segment Y. If segment Y flaps, and the port on BY becomes the alternate before sending out a single prestandard BPDU, AY cannot detect that a prestandard device is connected to Y and continues to send standard BPDUs. The port BY is fixed in a boundary, and no load balancing is possible between A and B. The same problem exists on segment X, but B might transmit topology



Note We recommend that you minimize the interaction between standard and prestandard MST implementations.

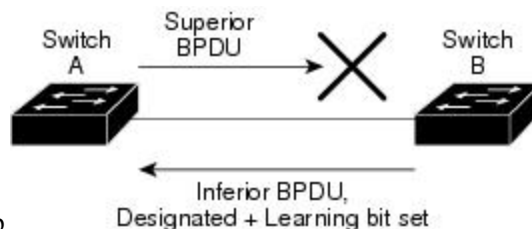
Detecting Unidirectional Link Failure

This feature is not yet present in the IEEE MST standard, but it is included in this Cisco IOS release. The software checks the consistency of the port role and state in the received BPDUs to detect unidirectional link failures that could cause bridging loops.

When a designated port detects a conflict, it keeps its role, but reverts to the discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 26: Detecting Unidirectional Link Failure

This figure illustrates a unidirectional link failure that typically creates a bridging loop. Device A is the root device, and its BPDUs are lost on the link leading to device B. RSTP and MST BPDUs include the role and state of the sending port. With this information, device A can detect that device B does not react to the superior BPDUs it sends and that device B is the designated, not root device. As a result, device A blocks (or keeps



blocking) its port, which prevents the bridging loop.

Interoperability with IEEE 802.1D STP

A device running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D devices. If this device receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP device also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MSTP BPDU (Version 3) associated with a different region, or an RSTP BPDU (Version 2).

However, the device does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy device has been removed from the link unless the legacy device is the designated device. A device might also continue to assign a boundary role to a port when the device to which this device is connected has joined the region. To restart the protocol migration process (force the renegotiation with neighboring devices), use the **clear spanning-tree detected-protocols** privileged EXEC command.

If all the legacy devices on the link are RSTP devices, they can process MSTP BPDUs as if they are RSTP BPDUs. Therefore, MSTP devices send either a Version 0 configuration and TCN BPDUs or Version 3 MSTP BPDUs on a boundary port. A boundary port connects to a LAN, the designated device of which is either a single spanning-tree device or a device with a different MST configuration.

RSTP Overview

The RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 1 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree).

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by learning the active topology. The RSTP builds upon the IEEE 802.1D STP to select the device with the highest device priority (lowest numerical priority value) as the root device. The RSTP then assigns one of these port roles to individual ports:

- Root port—Provides the best path (lowest cost) when the device forwards packets to the root device.
- Designated port—Connects to the designated device, which incurs the lowest path cost when forwarding packets from that LAN to the root device. The port through which the designated device is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root device to that provided by the current root port.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected in a loopback by a point-to-point link or when a device has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in IEEE 802.1D). The port state controls the operation of the forwarding and learning processes.

Table 45: Port State Comparison

Operational Status	STP Port State (IEEE 802.1D)	RSTP Port State	Is Port Included in the Active Topology?
Enabled	Blocking	Discarding	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

To be consistent with Cisco STP implementations, this guide defines the port state as *blocking* instead of *discarding*. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of a device, a device port, or a LAN. It provides rapid convergence for edge ports, new root ports, and ports connected through point-to-point links as follows:

- Edge ports—If you configure a port as an edge port on an RSTP device by using the **spanning-tree portfast** interface configuration command, the edge port immediately transitions to the forwarding state.

An edge port is the same as a Port Fast-enabled port, and you should enable it only on ports that connect to a single end station.

- Root ports—If the RSTP selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.
- Point-to-point links—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

Figure 27: Proposal and Agreement Handshaking for Rapid Convergence

Device A is connected to Device B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of Device A is a smaller numerical value than the priority of Device B. Device A sends a proposal message (a configuration BPDU with the proposal flag set) to Device B, proposing itself as the designated device.

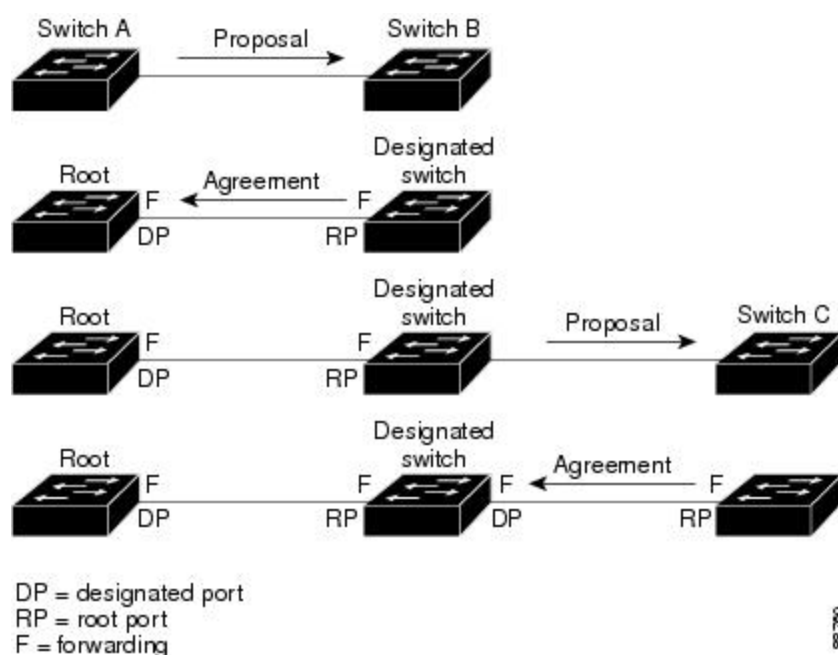
After receiving the proposal message, Device B selects as its new root port the port from which the proposal message was received, forces all nonedge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

After receiving Device B's agreement message, Device A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Device B blocked all of its nonedge ports and because there is a point-to-point link between Devices A and B.

When Device C is connected to Device B, a similar set of handshaking messages are exchanged. Device C selects the port connected to Device B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more device joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

In a device stack, the cross-stack rapid transition (CSRT) feature ensures that a stack member receives acknowledgments from all stack members during the proposal-agreement handshaking before moving the port to the forwarding state. CSRT is automatically enabled when the device is in MST mode.

The device learns the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. You can override the default setting that is controlled by the duplex setting by using the **spanning-tree link-type** interface configuration command.



Synchronization of Port Roles

When the device receives a proposal message on one of its ports and that port is selected as the new root port, the RSTP forces all other ports to synchronize with the new root information.

The device is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the device is synchronized if

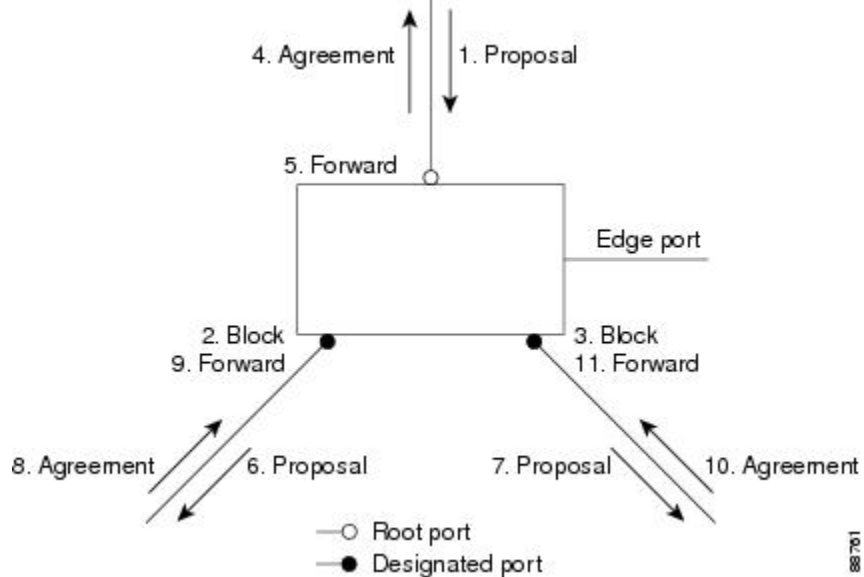
- That port is in the blocking state.
- It is an edge port (a port configured to be at the edge of the network).

If a designated port is in the forwarding state and is not configured as an edge port, it transitions to the blocking state when the RSTP forces it to synchronize with new root information. In general, when the RSTP forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

Figure 28: Sequence of Events During Rapid Convergence

After ensuring that all of the ports are synchronized, the device sends an agreement message to the designated device corresponding to its root port. When the devices connected by a point-to-point link are in agreement

about their port roles, the RSTP immediately transitions the port states to forwarding.



10488

Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new 1-byte Version 1 Length field is set to zero, which means that no version 1 protocol information is present.

Table 46: RSTP BPDU Flags

Bit	Function
0	Topology change (TC)
1	Proposal
2–3:	Port role:
00	Unknown
01	Alternate port
10	Root port
11	Designated port
4	Learning
5	Forwarding
6	Agreement
7	Topology change acknowledgement (TCA)

The sending device sets the proposal flag in the RSTP BPDU to propose itself as the designated device on that LAN. The port role in the proposal message is always set to the designated port.

The sending device sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with IEEE 802.1D devices, the RSTP device processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower device ID, lower path cost, and so forth) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the device sends an agreement message after all of the other ports are synchronized. If the BPDU is an IEEE 802.1D BPDU, the device does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (such as a higher device ID or a higher path cost than currently stored for the port) with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- **Detection**—Unlike IEEE 802.1D in which *any* transition between the blocking and the forwarding state causes a topology change, *only* transitions from the blocking to the forwarding state cause a topology change with RSTP (only an increase in connectivity is considered a topology change). State changes on an edge port do not cause a topology change. When an RSTP device detects a topology change, it deletes the learned information on all of its nonedge ports except on those from which it received the TC notification.
- **Notification**—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for IEEE 802.1D interoperability, an RSTP device processes and generates TCN BPDUs.
- **Acknowledgement**—When an RSTP device receives a TCN message on a designated port from an IEEE 802.1D device, it replies with an IEEE 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the topology-change timer in IEEE 802.1D) is active on a root port connected to an IEEE 802.1D device and a configuration BPDU with the TCA bit set is received, the TC-while timer is reset.

This behavior is only required to support IEEE 802.1D devices. The RSTP BPDUs never have the TCA bit set.

- **Propagation**—When an RSTP device receives a TC message from another device through a designated or root port, it propagates the change to all of its nonedge, designated ports and to the root port (excluding

the port on which it is received). The device starts the TC-while timer for all such ports and flushes the information learned on them.

- Protocol migration—For backward compatibility with IEEE 802.1D devices, RSTP selectively sends IEEE 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

When a port is initialized, the migrate-delay timer is started (specifies the minimum time during which RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the device processes all BPDUs received on that port and ignores the protocol type.

If the device receives an IEEE 802.1D BPDU after the port migration-delay timer has expired, it assumes that it is connected to an IEEE 802.1D device and starts using only IEEE 802.1D BPDUs. However, if the RSTP device is using IEEE 802.1D BPDUs on a port and receives an RSTP BPDU after the timer has expired, it restarts the timer and starts using RSTP BPDUs on that port.

Protocol Migration Process

A device running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D devices. If this device receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP device also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MST BPDU (Version 3) associated with a different region, or an RST BPDU (Version 2).

However, the device does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy device has been removed from the link unless the legacy device is the designated device. A device also might continue to assign a boundary role to a port when the device to which it is connected has joined the region.

Default MSTP Configuration

Table 47: Default MSTP Configuration

Feature	Default Setting
Spanning-tree mode	MSTP
Switch priority (configurable on a per-CIST port basis)	32768
Spanning-tree port priority (configurable on a per-CIST port basis)	128
Spanning-tree port cost (configurable on a per-CIST port basis)	1000 Mb/s: 20000 100 Mb/s: 20000 10 Mb/s: 20000
Hello time	3 seconds
Forward-delay time	20 seconds
Maximum-aging time	20 seconds
Maximum hop count	20 hops

About MST-to-PVST+ Interoperability (PVST+ Simulation)

The PVST+ simulation feature enables seamless interoperability between MST and Rapid PVST+. You can enable or disable this per port, or globally. PVST+ simulation is enabled by default.

However, you may want to control the connection between MST and Rapid PVST+ to protect against accidentally connecting an MST-enabled port to a Rapid PVST+-enabled port. Because Rapid PVST+ is the default STP mode, you may encounter many Rapid PVST+-enabled connections.

Disabling this feature causes the switch to stop the MST region from interacting with PVST+ regions. The MST-enabled port moves to a PVST peer inconsistent (blocking) state once it detects it is connected to a Rapid PVST+-enabled port. This port remains in the inconsistent state until the port stops receiving Shared Spanning Tree Protocol (SSTP) BPDUs, and then the port resumes the normal STP transition process.

You can for instance, disable PVST+ simulation, to prevent an incorrectly configured switch from connecting to a network where the STP mode is not MSTP (the default mode is PVST+).

Observe these guidelines when you configure MST switches (in the same region) to interact with PVST+ switches:

- Configure the root for all VLANs inside the MST region as shown in this example:

```
Switch# show spanning-tree mst interface gigabitethernet 1/1
GigabitEthernet1/1 of MST00 is root forwarding
Edge port: no (trunk) port guard : none (default)
Link type: point-to-point (auto) bpdu filter: disable (default)
Boundary : boundary (PVST) bpdu guard : disable (default)
Bpdus sent 10, received 310
```

Instance	Role	Sts	Cost	Prio.Nbr	Vlans mapped
0	Root	FWD	20000	128.1	1-2, 4-2999, 4000-4094
3	Boun	FWD	20000	128.1	3, 3000-3999

The ports that belong to the MST switch at the boundary simulate PVST+ and send PVST+ BPDUs for all the VLANs.

If you enable loop guard on the PVST+ switches, the ports might change to a loop-inconsistent state when the MST switches change their configuration. To correct the loop-inconsistent state, you must disable and re-enable loop guard on that PVST+ switch.

- Do not locate the root for some or all of the VLANs inside the PVST+ side of the MST switch because when the MST switch at the boundary receives PVST+ BPDUs for all or some of the VLANs on its designated ports, root guard sets the port to the blocking state.
- When you connect a PVST+ switch to two different MST regions, the topology change from the PVST+ switch does not pass beyond the first MST region. In such a case, the topology changes are propagated only in the instance to which the VLAN is mapped. The topology change stays local to the first MST region, and the Cisco Access Manager (CAM) entries in the other region are not flushed. To make the topology change visible throughout other MST regions, you can map that VLAN to IST or connect the PVST+ switch to the two regions through access links.
- When you disable the PVST+ simulation, note that the PVST+ peer inconsistency can also occur while the port is already in other states of inconsistency. For example, the root bridge for all STP instances must all be in either the MST region or the Rapid PVST+ side. If the root bridge for all STP instances are not on one side or the other, the software moves the port into a PVST + simulation-inconsistent state.



Note We recommend that you put the root bridge for all STP instances in the MST region.

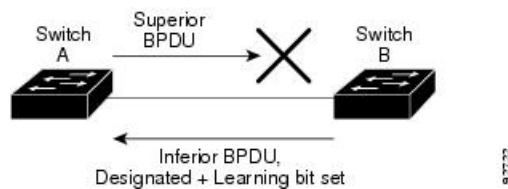
About Detecting Unidirectional Link Failure

The dispute mechanism that detects unidirectional link failures is included in the IEEE 802.1D-2004 RSTP and IEEE 802.1Q-2005 MSTP standard, and requires no user configuration.

The switch checks the consistency of the port role and state in the BPDUs it receives, to detect unidirectional link failures that could cause bridging loops. When a designated port detects a conflict, it keeps its role, but reverts to a discarding (blocking) state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

For example, in the figure below, Switch A is the root bridge and Switch B is the designated port. BPDUs from Switch A are lost on the link leading to switch B.

Figure 29: Detecting Unidirectional Link Failure

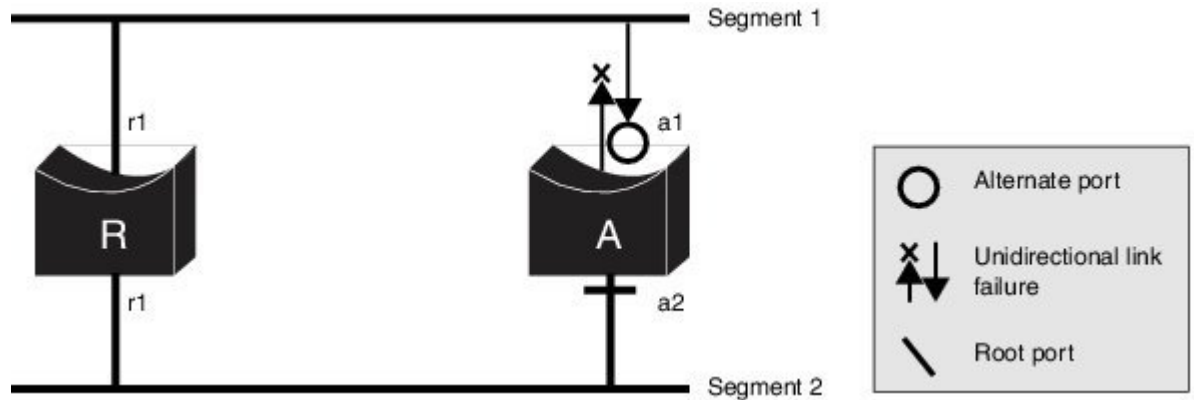


Since Rapid PVST+ (802.1w) and MST BPDUs include the role and state of the sending port, Switch A detects (from the inferior BPDU), that switch B does not react to the superior BPDUs it sends, because switch B has the role of a designated port and not the root bridge. As a result, switch A blocks (or keeps blocking) its port, thus preventing the bridging loop.

Note these guidelines and limitations relating to the dispute mechanism:

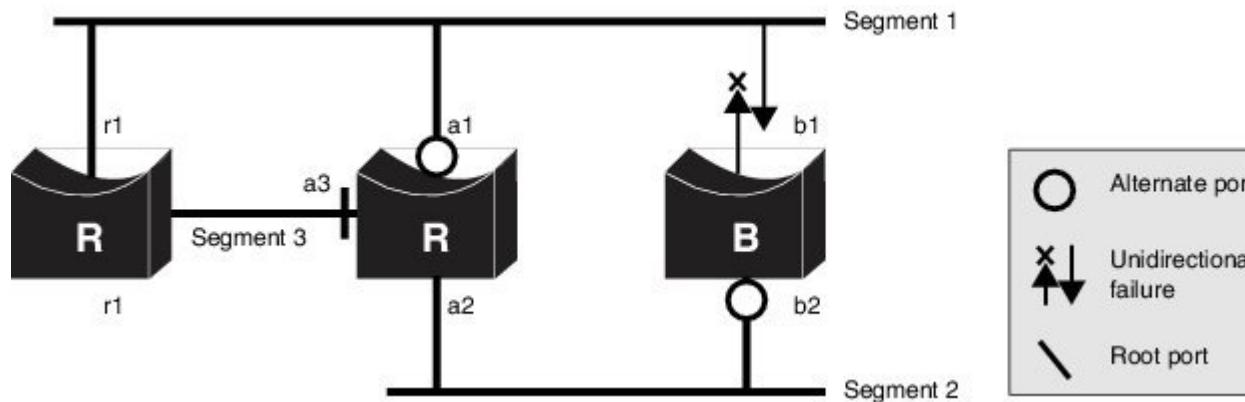
- It works only on switches running RSTP or MST (the dispute mechanism requires reading the role and state of the port initiating BPDUs).
- It may result in loss of connectivity. For example, in the figure below, Bridge A cannot transmit on the port it elected as a root port. As a result of this situation, there is loss of connectivity (r1 and r2 are designated, a1 is root and a2 is alternate. There is only a one way connectivity between A and R).

Figure 30: Loss of Connectivity



- It may cause permanent bridging loops on shared segments. For example, in the figure below, suppose that bridge R has the best priority, and that port b1 cannot receive any traffic from the shared segment 1 and sends inferior designated information on segment 1. Both r1 and a1 can detect this inconsistency. However, with the current dispute mechanism, only r1 will revert to discarding while the root port a1 opens a permanent loop. However, this problem does not occur in Layer 2 switched networks that are connected by point-to-point links.

Figure 31: Bridging Loops on Shared Segments



How to Configure MSTP Features

Specifying the MST Region Configuration and Enabling MSTP

For two or more switches to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst configuration**
4. **instance** *instance-id* **vlan** *vlan-range*
5. **name** *name*
6. **revision** *version*
7. **show pending**
8. **exit**
9. **spanning-tree mode mst**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst configuration Example: Switch(config)# spanning-tree mst configuration	Enters MST configuration mode.
Step 4	instance <i>instance-id</i> vlan <i>vlan-range</i> Example: Switch(config-mst)# instance 1 vlan 10-20	Maps VLANs to an MST instance. <ul style="list-style-type: none"> • For <i>instance-id</i>, the range is 0 to 4094. • For vlan <i>vlan-range</i>, the range is 1 to 4094. When you map VLANs to an MST instance, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped. <p>To specify a VLAN range, use a hyphen; for example, instance 1 vlan 1-63 maps VLANs 1 through 63 to MST instance 1.</p> <p>To specify a VLAN series, use a comma; for example, instance 1 vlan 10, 20, 30 maps VLANs 10, 20, and 30 to MST instance 1.</p>

	Command or Action	Purpose
Step 5	name <i>name</i> Example: <pre>Switch(config-mst) # name region1</pre>	Specifies the configuration name. The <i>name</i> string has a maximum length of 32 characters and is case sensitive.
Step 6	revision <i>version</i> Example: <pre>Switch(config-mst) # revision 1</pre>	Specifies the configuration revision number. The range is 0 to 65535.
Step 7	show pending Example: <pre>Switch(config-mst) # show pending</pre>	Verifies your configuration by displaying the pending configuration.
Step 8	exit Example: <pre>Switch(config-mst) # exit</pre>	Applies all changes, and returns to global configuration mode.
Step 9	spanning-tree mode mst Example: <pre>Switch(config) # spanning-tree mode mst</pre>	<p>Enables MSTP. RSTP is also enabled.</p> <p>Changing spanning-tree modes can disrupt traffic because all spanning-tree instances are stopped for the previous mode and restarted in the new mode.</p> <p>You cannot run both MSTP and PVST+ or both MSTP and Rapid PVST+ at the same time.</p>
Step 10	end Example: <pre>Switch(config) # end</pre>	Returns to privileged EXEC mode.

Configuring the Root Device

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID. Step 2 in the example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `spanning-tree mst instance-id root primary`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree mst <i>instance-id</i> root primary Example: <pre>Switch(config)# spanning-tree mst 0 root primary</pre>	Configures a device as the root device. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring a Secondary Root Device

When you configure a device with the extended system ID support as the secondary root, the device priority is modified from the default value (32768) to 28672. The device is then likely to become the root device for the specified instance if the primary root device fails. This is assuming that the other network devices use the default device priority of 32768 and therefore are unlikely to become the root device.

You can execute this command on more than one device to configure multiple backup root devices. Use the same network diameter and hello-time values that you used when you configured the primary root device with the **spanning-tree mst *instance-id* root primary** global configuration command.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst *instance-id* root secondary**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst <i>instance-id</i> root secondary Example: Switch(config)# spanning-tree mst 0 root secondary	Configures a device as the secondary root device. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Configuring Port Priority

If a loop occurs, the MSTP uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.



Note If the device is a member of a device stack, you must use the **spanning-tree mst [instance-id] cost cost** interface configuration command instead of the **spanning-tree mst [instance-id] port-priority priority** interface configuration command to select a port to put in the forwarding state. Assign lower cost values to ports that you want selected first and higher cost values to ports that you want selected last. For more information, see the path costs topic listed under Related Topics.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **spanning-tree mst instance-id port-priority priority**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies an interface to configure, and enters interface configuration mode.
Step 4	spanning-tree mst instance-id port-priority priority Example:	Configures port priority. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series

	Command or Action	Purpose
	<pre>Switch(config-if) # spanning-tree mst 0 port-priority 64</pre>	<p>of instances separated by a comma. The range is 0 to 4094.</p> <ul style="list-style-type: none"> For <i>priority</i>, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. <p>The priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-if) # end</pre>	Returns to privileged EXEC mode.

The **show spanning-tree mst interface *interface-id*** privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

Configuring Path Cost

The MSTP path cost default value is derived from the media speed of an interface. If a loop occurs, the MSTP uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **spanning-tree mst *instance-id* cost *cost***
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces. The port-channel range is 1 to 48.
Step 4	spanning-tree mst <i>instance-id</i> cost <i>cost</i> Example: <pre>Switch(config-if)# spanning-tree mst 0 cost 17031970</pre>	Configures the cost. <p>If a loop occurs, the MSTP uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission.</p> <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For <i>cost</i>, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

The **show spanning-tree mst interface *interface-id*** privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

Configuring the Device Priority

Changing the priority of a device makes it more likely to be chosen as the root device whether it is a standalone device or a device in the stack.



Note Exercise care when using this command. For normal network configurations, we recommend that you use the **spanning-tree mst *instance-id* root primary** and the **spanning-tree mst *instance-id* root secondary** global configuration commands to specify a device as the root or secondary root device. You should modify the device priority only in circumstances where these commands do not work.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID used. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst *instance-id* priority *priority***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst <i>instance-id</i> priority <i>priority</i> Example: Switch(config)# spanning-tree mst 0 priority 40960	Configures the device priority. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For <i>priority</i>, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the device will be chosen as the root device.

	Command or Action	Purpose
		Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. These are the only acceptable values.
Step 4	end Example: Switch(config-if) # end	Returns to privileged EXEC mode.

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root device.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst hello-time** *seconds*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst hello-time <i>seconds</i> Example: Switch(config)# spanning-tree mst hello-time 4	Configures the hello time for all MST instances. The hello time is the time interval between configuration messages generated and sent by the root device. These messages indicate that the device is alive.

	Command or Action	Purpose
		For <i>seconds</i> , the range is 1 to 10; the default is 3.
Step 4	end Example: Switch(config) # end	Returns to privileged EXEC mode.

Configuring the Forwarding-Delay Time

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst forward-time *seconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst forward-time <i>seconds</i> Example: Switch(config) # spanning-tree mst forward-time 25	Configures the forward time for all MST instances. The forwarding delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For <i>seconds</i> , the range is 4 to 30; the default is 20.
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	<code>Switch(config)# end</code>	

Configuring the Maximum-Aging Time

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `spanning-tree mst max-age seconds`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>spanning-tree mst max-age seconds</code></p> <p>Example:</p> <pre>Switch(config)# spanning-tree mst max-age 40</pre>	<p>Configures the maximum-aging time for all MST instances. The maximum-aging time is the number of seconds a device waits without receiving spanning-tree configuration messages before attempting a reconfiguration.</p> <p>For <i>seconds</i>, the range is 6 to 40; the default is 20.</p>
Step 4	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Configuring the Maximum-Hop Count

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-hops *hop-count***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst max-hops <i>hop-count</i> Example: Switch(config)# spanning-tree mst max-hops 25	Specifies the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. For <i>hop-count</i> , the range is 1 to 255; the default is 20.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Specifying the Link Type to Ensure Rapid Transitions

If you connect a port to another port through a point-to-point link and the local port becomes a designated port, the RSTP negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

By default, the link type is controlled from the duplex mode of the interface: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. If you have a half-duplex link physically connected point-to-point to a single port on a remote device running MSTP, you can override the default setting of the link type and enable rapid transitions to the forwarding state.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **spanning-tree link-type point-to-point**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port-channel logical interfaces. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48.
Step 4	spanning-tree link-type point-to-point Example: <pre>Switch(config-if)# spanning-tree link-type point-to-point</pre>	Specifies that the link type of a port is point-to-point.

	Command or Action	Purpose
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Designating the Neighbor Type

A topology could contain both prestandard and IEEE 802.1s standard compliant devices. By default, ports can automatically detect prestandard devices, but they can still receive both standard and prestandard BPDUs. When there is a mismatch between a device and its neighbor, only the CIST runs on the interface.

You can choose to set a port to send only prestandard BPDUs. The prestandard flag appears in all the **show** commands, even if the port is in STP compatibility mode.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst pre-standard**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports.

	Command or Action	Purpose
Step 4	spanning-tree mst pre-standard Example: <pre>Switch(config-if)# spanning-tree mst pre-standard</pre>	Specifies that the port can send only prestandard BPDUs.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Restarting the Protocol Migration Process

This procedure restarts the protocol migration process and forces renegotiation with neighboring devices. It reverts the device to MST mode. It is needed when the device no longer receives IEEE 802.1D BPDUs after it has been receiving them.

Follow these steps to restart the protocol migration process (force the renegotiation with neighboring devices) on the device.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

If you want to use the interface version of the command, you must also know the MST interface used. This example uses `GigabitEthernet1/0/1` as the interface because that was the interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. Enter one of the following commands:
 - **clear spanning-tree detected-protocols**
 - **clear spanning-tree detected-protocols interface** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • clear spanning-tree detected-protocols 	The device reverts to the MSTP mode, and the protocol migration process restarts.

	Command or Action	Purpose
	<ul style="list-style-type: none"> • <code>clear spanning-tree detected-protocols interface interface-id</code> <p>Example:</p> <pre>Switch# clear spanning-tree detected-protocols or Switch# clear spanning-tree detected-protocols interface gigabitethernet 1/0/1</pre>	

What to do next

This procedure may need to be repeated if the device receives more legacy IEEE 802.1D configuration BPDUs (BPDUs with the protocol version set to 0).

Configuring PVST+ Simulation

PVST+ simulation is enabled by default. This means that all ports automatically interoperate with a connected device that is running in Rapid PVST+ mode. If you disabled the feature and want to re-configure it, refer to the following tasks.

To enable PVST+ simulation globally, perform this task:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `spanning-tree mst simulate pvst global`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>spanning-tree mst simulate pvst global</code></p> <p>Example:</p>	<p>Enables PVST+ simulation globally.</p> <p>To prevent the switch from automatically interoperating with a connecting switch that is running Rapid PVST+, enter the no version of the command.</p>

	Command or Action	Purpose
	<code>Switch(config)# spanning-tree mst simulate pvst global</code>	
Step 4	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.

Enabling PVST+ Simulation on a Port

To enable PVST+ simulation on a port, perform this task:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `spanning-tree mst simulate pvst`
5. `end`
6. `show spanning-tree summary`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <code>Switch> enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 3	interface interface-id Example: <code>Switch(config)# interface gi1/0/1</code>	Selects a port to configure.
Step 4	spanning-tree mst simulate pvst Example:	Enables PVST+ simulation on the specified interface. To prevent a specified interface from automatically interoperating with a connecting switch that is not running

	Command or Action	Purpose
	Switch(config-if) # spanning-tree mst simulate pvst	MST, enter the spanning-tree mst simulate pvst disable command.
Step 5	end Example: Switch(config) # end	Returns to privileged EXEC mode.
Step 6	show spanning-tree summary Example: Switch# show spanning-tree summary	Verifies the configuration.

Examples

Examples: PVST+ Simulation

This example shows how to prevent the switch from automatically interoperating with a connecting switch that is running Rapid PVST+:

```
Switch# configure terminal
Switch(config)# no spanning-tree mst simulate pvst global
```

This example shows how to prevent a port from automatically interoperating with a connecting device that is running Rapid PVST+:

```
Switch(config)# interface1/0/1
Switch(config-if)# spanning-tree mst simulate pvst disable
```

The following sample output shows the system message you receive when a SSTP BPDU is received on a port and PVST+ simulation is disabled:

```
Message
SPANTREE_PVST_PEER_BLOCK: PVST BPDU detected on port %s [port number].
```

```
Severity
Critical
```

```
Explanation
A PVST+ peer was detected on the specified interface on the switch. PVST+
simulation feature is disabled, as a result of which the interface was
moved to the spanning tree
Blocking state.
```

```
Action
```

Identify the PVST+ switch from the network which might be configured incorrectly.

The following sample output shows the system message you receive when peer inconsistency on the interface is cleared:

```
Message
SPANTREE_PVST_PEER_UNBLOCK: Unblocking port %s [port number].
```

```
Severity
Critical
```

```
Explanation
The interface specified in the error message has been restored to normal
spanning tree state.
```

```
Action
None.
```

This example shows the spanning tree status when port **1/0/1** has been configured to disable PVST+ simulation and is currently in the peer type inconsistent state:

```
Switch# show spanning-tree
VLAN0010
  Spanning tree enabled protocol mstp
  Root ID Priority 32778
        Address 0002.172c.f400
        This bridge is the root
        Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
        Address 0002.172c.f400
        Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
        Aging Time 300
Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/1       Desg BKN*4    128.270 P2p *PVST_Peer_Inc
```

This example shows the spanning tree summary when PVST+ simulation is enabled in the MSTP mode:

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is enabled
Name          Blocking Listening Learning Forwarding STP Active
-----
MST0          2          0          0          0          2
```

```
-----
1 mst                2                0                0                0                2
```

This example shows the spanning tree summary when PVST+ simulation is disabled in any STP mode:

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is disabled
```

Name	Blocking	Listening	Learning	Forwarding	STP Active
MST0	2	0	0	0	2
1 mst	2	0	0	0	2

This example shows the spanning tree summary when the switch is not in MSTP mode, that is, the switch is in PVST or Rapid-PVST mode. The output string displays the current STP mode:

```
Switch# show spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001, VLAN2001-VLAN2002
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is short
PVST Simulation Default is enabled but inactive in rapid-pvst mode
```

Name	Blocking	Listening	Learning	Forwarding	STP Active
VLAN0001	2	0	0	0	2
VLAN2001	2	0	0	0	2
VLAN2002	2	0	0	0	2
3 vlans	6	0	0	0	6

This example shows the interface details when PVST+ simulation is globally enabled, or the default configuration:

```
Switch# show spanning-tree interface1/0/1 detail
```

```
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is enabled by default
  BPDU: sent 132, received 1
```

This example shows the interface details when PVST+ simulation is globally disabled:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is disabled by default
  BPDU: sent 132, received 1
```

This example shows the interface details when PVST+ simulation is explicitly enabled on the port:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is enabled
  BPDU: sent 132, received 1
```

This example shows the interface details when the PVST+ simulation feature is disabled and a PVST Peer inconsistency has been detected on the port:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is broken (PVST Peer Inconsistent)
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is disabled
  BPDU: sent 132, received 1
```

Examples: Detecting Unidirectional Link Failure

This example shows the spanning tree status when port `1/0/1 detail` has been configured to disable PVST+ simulation and the port is currently in the peer type inconsistent state:

```
Switch# show spanning-tree
VLAN0010
```

```

Spanning tree enabled protocol rstp
Root ID    Priority 32778
           Address 0002.172c.f400
           This bridge is the root
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID  Priority 32778 (priority 32768 sys-id-ext 10)
           Address 0002.172c.f400
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
           Aging Time 300

```

```

Interface          Role Sts Cost          Prio.Nbr Type
-----
Gi1/0/1            Desg BKN 4          128.270 P2p Dispute

```

This example shows the interface details when a dispute condition is detected:

```

Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is designated blocking (dispute)
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  BPDU: sent 132, received 1

```

Monitoring MST Configuration and Status

Table 48: Commands for Displaying MST Status

show spanning-tree mst configuration	Displays the MST region configuration.
show spanning-tree mst configuration digest	Displays the MD5 digest included in the current MSTCI.
show spanning-tree mst <i>instance-id</i>	Displays MST information for the specified instance. Note This command displays information only if the port is in operative state.
show spanning-tree mst interface <i>interface-id</i>	Displays MST information for the specified interface.

Feature Information for MSTP

Release	Modification
Cisco IOS Release 15.2(3)E	This feature was introduced.



CHAPTER 26

Configuring Optional Spanning-Tree Features

- [Finding Feature Information, on page 481](#)
- [Restriction for Optional Spanning-Tree Features, on page 481](#)
- [Information About Optional Spanning-Tree Features, on page 481](#)
- [How to Configure Optional Spanning-Tree Features, on page 493](#)
- [Examples, on page 508](#)
- [Monitoring the Spanning-Tree Status, on page 511](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restriction for Optional Spanning-Tree Features

- PortFast minimizes the time that interfaces must wait for spanning tree to converge, so it is effective only when used on interfaces connected to end stations. If you enable PortFast on an interface connecting to another switch, you risk creating a spanning-tree loop.

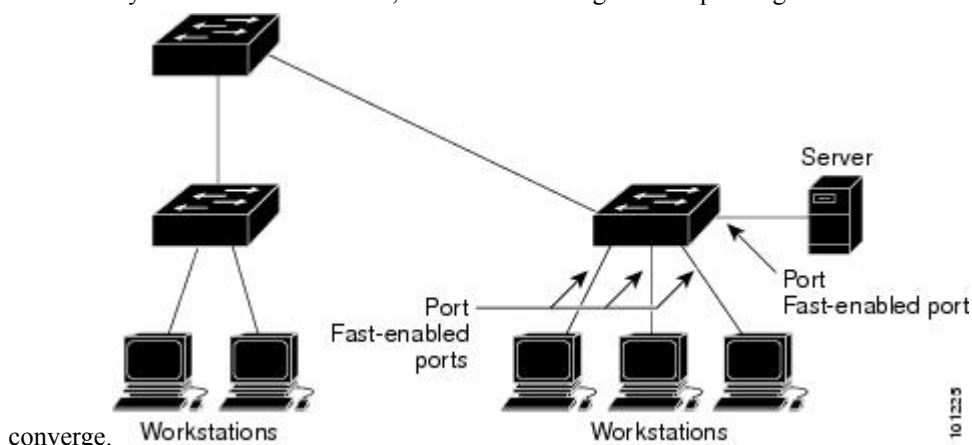
Information About Optional Spanning-Tree Features

PortFast

PortFast immediately brings an interface configured as an access or trunk port to the forwarding state from a blocking state, bypassing the listening and learning states.

Figure 32: PortFast-Enabled Interfaces

You can use PortFast on interfaces connected to a single workstation or server to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to



Interfaces connected to a single workstation or server should not receive bridge protocol data units (BPDUs). An interface with PortFast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

You can enable this feature by enabling it on either the interface or on all nontrunking ports.

BPDU Guard

The Bridge Protocol Data Unit (BPDU) guard feature can be globally enabled on the switch or can be enabled per port, but the feature operates with some differences.

When you enable BPDU guard at the global level on PortFast edge-enabled ports, spanning tree shuts down ports that are in a PortFast edge-operational state if any BPDU is received on them. In a valid configuration, PortFast edge-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast edge-enabled port means an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. When this happens, the switch shuts down the entire port on which the violation occurred.

When you enable BPDU guard at the interface level on any port without also enabling the PortFast edge feature, and the port receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

BPDU Filtering

The BPDU filtering feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

Enabling BPDU filtering on PortFast edge-enabled interfaces at the global level keeps those interfaces that are in a PortFast edge-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these interfaces do not receive BPDUs. If a BPDU is received

on a PortFast edge-enabled interface, the interface loses its PortFast edge-operational status, and BPDU filtering is disabled.

Enabling BPDU filtering on an interface without also enabling the PortFast edge feature keeps the interface from sending or receiving BPDUs.



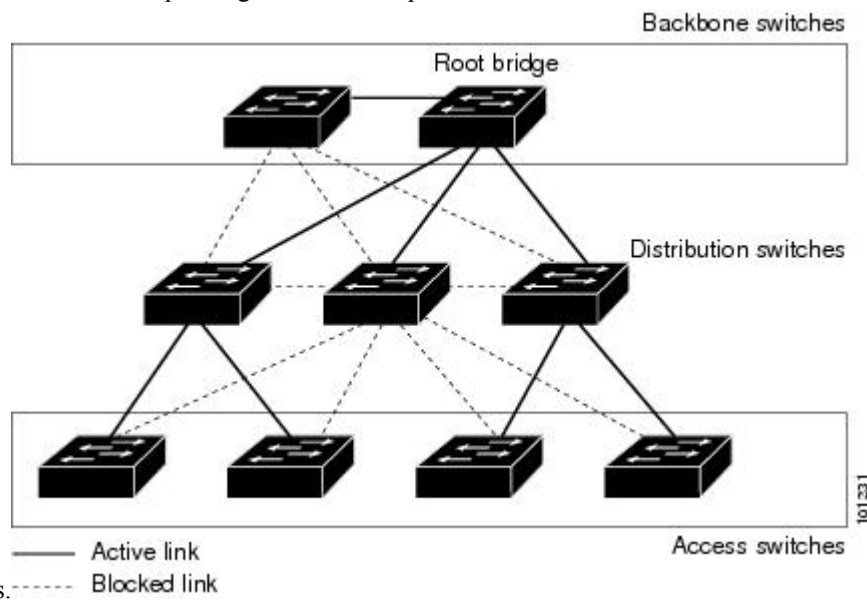
Caution Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire switch or for an interface.

UplinkFast

Figure 33: Switches in a Hierarchical Network

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. This complex network has distribution switches and access switches that each have at least one redundant link that spanning tree blocks to prevent loops.



If a switch loses connectivity, it begins using the alternate paths as soon as the spanning tree selects a new root port. You can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself by enabling UplinkFast. The root port transitions to the forwarding state immediately without going through the listening and learning states, as it would with the normal spanning-tree procedures.

When the spanning tree reconfigures the new root port, other interfaces flood the network with multicast packets, one for each address that was learned on the interface. You can limit these bursts of multicast traffic by reducing the max-update-rate parameter (the default for this parameter is 150 packets per second). However, if you enter zero, station-learning frames are not generated, so the spanning-tree topology converges more slowly after a loss of connectivity.

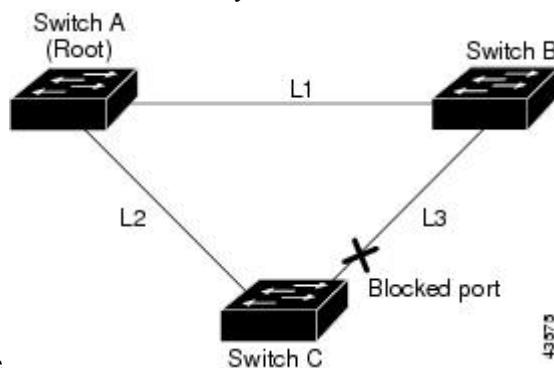


Note UplinkFast is most useful in wiring-closet switches at the access or edge of the network. It is not appropriate for backbone devices. This feature might not be useful for other types of applications.

UplinkFast provides fast convergence after a direct link failure and achieves load-balancing between redundant Layer 2 links using uplink groups. An uplink group is a set of Layer 2 interfaces (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports, except for self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

Figure 34: UplinkFast Example Before Direct Link Failure

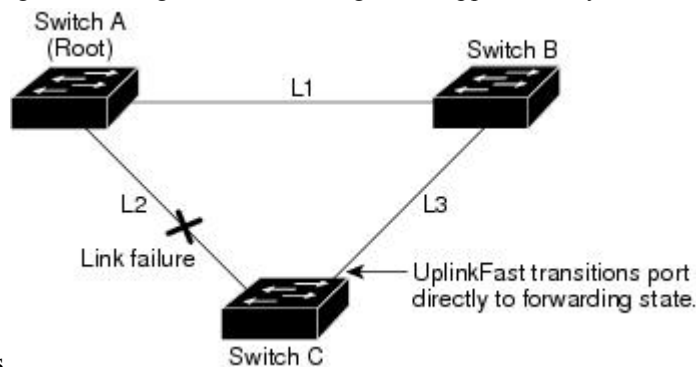
This topology has no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in



a blocking state.

Figure 35: UplinkFast Example After Direct Link Failure

If Switch C detects a link failure on the currently active link L2 on the root port (a direct link failure), UplinkFast unblocks the blocked interface on Switch C and transitions it to the forwarding state without going through the listening and learning states. This change takes approximately 1 to



5 seconds.

Cross-Stack UplinkFast

Cross-Stack UplinkFast (CSUF) provides a fast spanning-tree transition (fast convergence in less than 1 second under normal network conditions) across a switch stack. During the fast transition, an alternate redundant link on the switch stack is placed in the forwarding state without causing temporary spanning-tree loops or loss of connectivity to the backbone. With this feature, you can have a redundant and resilient network in some configurations. CSUF is automatically enabled when you enable the UplinkFast feature.

CSUF might not provide a fast transition all the time; in these cases, the normal spanning-tree transition occurs, completing in 30 to 40 seconds. For more information, see [Related Topics](#).

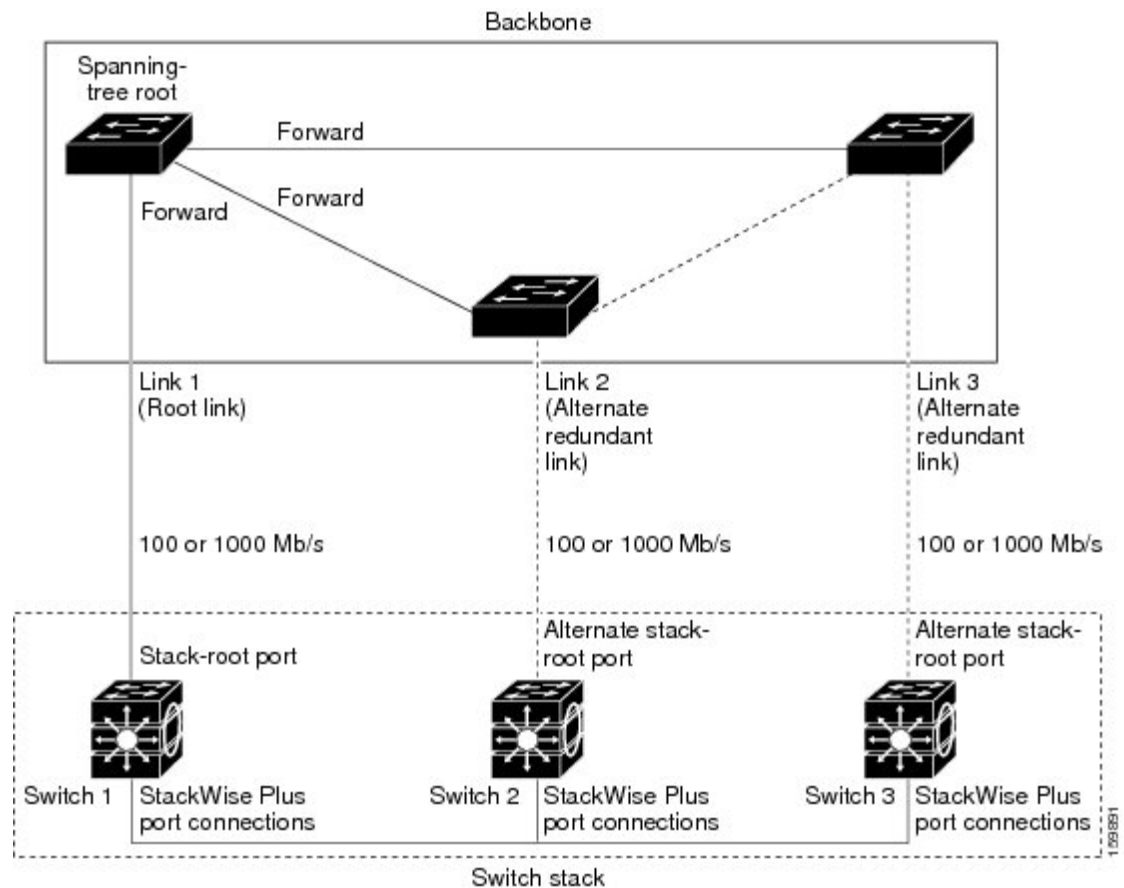
How Cross-Stack UplinkFast Works

Cross-Stack UplinkFast (CSUF) ensures that one link in the stack is elected as the path to the root.

Figure 36: Cross-Stack UplinkFast Topology

The stack-root port on Switch 1 provides the path to the root of the spanning tree. The alternate stack-root ports on Switches 2 and 3 can provide an alternate path to the spanning-tree root if the current stack-root switch fails or if its link to the spanning-tree root fails.

Link 1, the root link, is in the spanning-tree forwarding state. Links 2 and 3 are alternate redundant links that are in the spanning-tree blocking state. If Switch 1 fails, if its stack-root port fails, or if Link 1 fails, CSUF selects either the alternate stack-root port on Switch 2 or Switch 3 and puts it into the forwarding state in less than 1 second.



When certain link loss or spanning-tree events occur (described in the following topic), the Fast Uplink Transition Protocol uses the neighbor list to send fast-transition requests to stack members.

The switch sending the fast-transition request needs to do a fast transition to the forwarding state of a port that it has chosen as the root port, and it must obtain an acknowledgment from each stack switch before performing the fast transition.

Each switch in the stack decides if the sending switch is a better choice than itself to be the stack root of this spanning-tree instance by comparing the root, cost, and bridge ID. If the sending switch is the best choice as the stack root, each switch in the stack returns an acknowledgment; otherwise, it sends a fast-transition request. The sending switch then has not received acknowledgments from all stack switches.

When acknowledgments are received from all stack switches, the Fast Uplink Transition Protocol on the sending switch immediately transitions its alternate stack-root port to the forwarding state. If acknowledgments from all stack switches are not obtained by the sending switch, the normal spanning-tree transitions (blocking, listening, learning, and forwarding) take place, and the spanning-tree topology converges at its normal rate ($2 * \text{forward-delay time} + \text{max-age time}$).

The Fast Uplink Transition Protocol is implemented on a per-VLAN basis and affects only one spanning-tree instance at a time.

Events That Cause Fast Convergence

Depending on the network event or failure, the CSUF fast convergence might or might not occur.

Fast convergence (less than 1 second under normal network conditions) occurs under these circumstances:

- The stack-root port link fails.
If two switches in the stack have alternate paths to the root, only one of the switches performs the fast transition.
- The failed link, which connects the stack root to the spanning-tree root, recovers.
- A network reconfiguration causes a new stack-root switch to be selected.
- A network reconfiguration causes a new port on the current stack-root switch to be chosen as the stack-root port.



Note The fast transition might not occur if multiple events occur simultaneously. For example, if a stack member is powered off, and at the same time, the link connecting the stack root to the spanning-tree root comes back up, the normal spanning-tree convergence occurs.

Normal spanning-tree convergence (30 to 40 seconds) occurs under these conditions:

- The stack-root switch is powered off, or the software failed.
- The stack-root switch, which was powered off or failed, is powered on.
- A new switch, which might become the stack root, is added to the stack.

BackboneFast

BackboneFast detects indirect failures in the core of the backbone. BackboneFast is a complementary technology to the UplinkFast feature, which responds to failures on links directly connected to access switches.

BackboneFast optimizes the maximum-age timer, which controls the amount of time the switch stores protocol information received on an interface. When a switch receives an inferior BPDU from the designated port of another switch, the BPDU is a signal that the other switch might have lost its path to the root, and BackboneFast tries to find an alternate path to the root.

BackboneFast starts when a root port or blocked interface on a switch receives inferior BPDUs from its designated switch. An inferior BPDU identifies a switch that declares itself as both the root bridge and the designated switch. When a switch receives an inferior BPDU, it means that a link to which the switch is not directly connected (an indirect link) has failed (that is, the designated switch has lost its connection to the root switch). Under spanning-tree rules, the switch ignores inferior BPDUs for the maximum aging time (default is 20 seconds).

The switch tries to find if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked interface, the root port and other blocked interfaces on the switch become alternate paths to the root switch. (Self-looped ports are not considered alternate paths to the root switch.) If the inferior BPDU arrives on the root port, all blocked interfaces become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked interfaces, the switch assumes that it has lost connectivity to the root switch, causes the maximum aging time on the root port to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root switch, it uses these alternate paths to send a root link query (RLQ) request. The switch sends the RLQ request on all alternate paths to learn if any stack member has an alternate root to the root switch and waits for an RLQ reply from other switches in the network and in the stack. The switch sends the RLQ request on all alternate paths and waits for an RLQ reply from other switches in the network.

When a stack member receives an RLQ reply from a nonstack member on a blocked interface and the reply is destined for another nonstacked switch, it forwards the reply packet, regardless of the spanning-tree interface state.

When a stack member receives an RLQ reply from a nonstack member and the response is destined for the stack, the stack member forwards the reply so that all the other stack members receive it.

If the switch discovers that it still has an alternate path to the root, it expires the maximum aging time on the interface that received the inferior BPDU. If all the alternate paths to the root switch indicate that the switch has lost connectivity to the root switch, the switch expires the maximum aging time on the interface that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all interfaces on which it received an inferior BPDU its designated ports and moves them from the blocking state (if they were in the blocking state), through the listening and learning states, and into the forwarding state.

Figure 37: BackboneFast Example Before Indirect Link Failure

This is an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that connects directly to Switch B

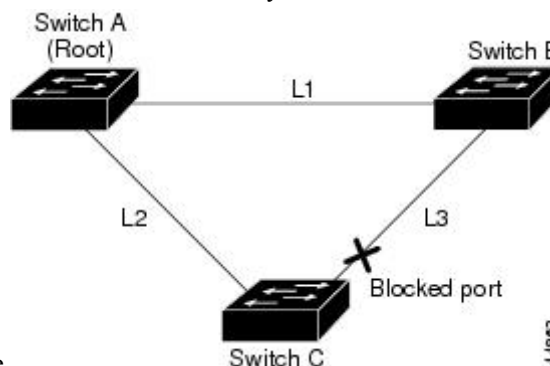


Figure 38: BackboneFast Example After Indirect Link Failure

If link L1 fails, Switch C cannot detect this failure because it is not connected directly to link L1. However, because Switch B is directly connected to the root switch over L1, it detects the failure, elects itself the root, and begins sending BPDUs to Switch C, identifying itself as the root. When Switch C receives the inferior BPDUs from Switch B, Switch C assumes that an indirect failure has occurred. At that point, BackboneFast allows the blocked interface on Switch C to move immediately to the listening state without waiting for the maximum aging time for the interface to expire. BackboneFast then transitions the Layer 2 interface on Switch C to the forwarding state, providing a path from Switch B to Switch A. The root-switch election takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. BackboneFast reconfigures the topology to account for the failure of link

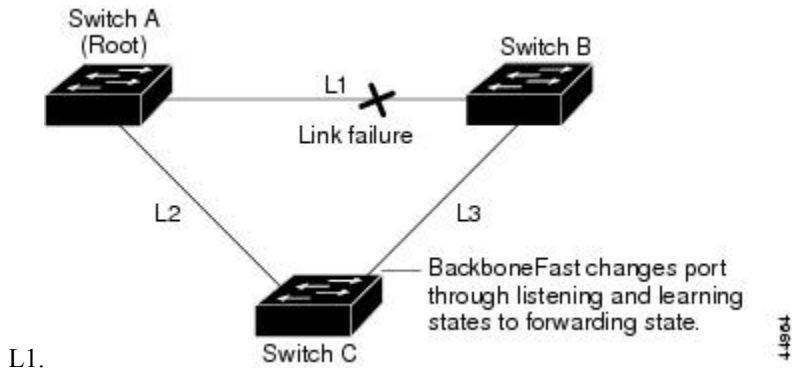
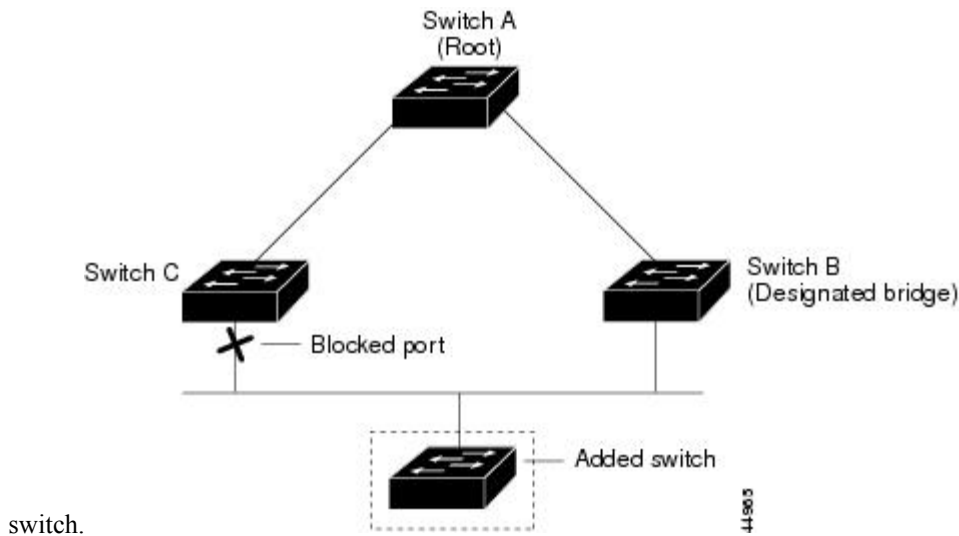


Figure 39: Adding a Switch in a Shared-Medium Topology

If a new switch is introduced into a shared-medium topology, BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated switch (Switch B). The new switch begins sending inferior BPDUs that indicate it is the root switch. However, the other switches ignore these inferior BPDUs, and the new switch learns that Switch B is the designated switch to Switch A, the root



EtherChannel Guard

You can use EtherChannel guard to detect an EtherChannel misconfiguration between the switch and a connected device. A misconfiguration can occur if the switch interfaces are configured in an EtherChannel,

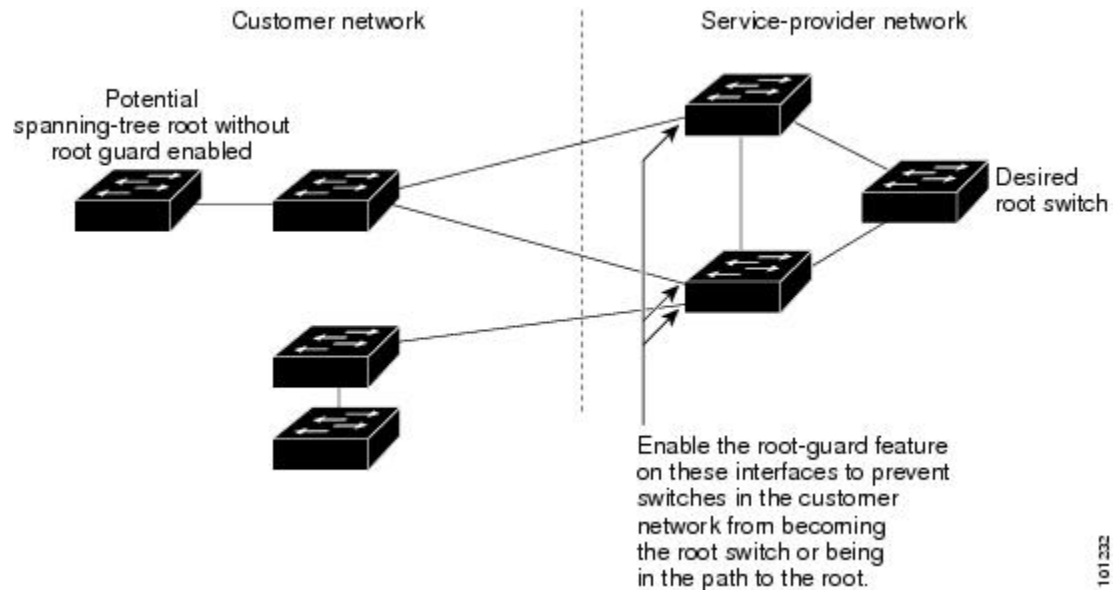
but the interfaces on the other device are not. A misconfiguration can also occur if the channel parameters are not the same at both ends of the EtherChannel.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and displays an error message.

Root Guard

Figure 40: Root Guard in a Service-Provider Network

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a customer switch as the root switch. You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.



If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the interface to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the interface also is blocked in all MST instances. A boundary port is an interface that connects to a LAN, the designated switch of which is either an IEEE 802.1D switch or a switch with a different MST region configuration.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.



Caution Misuse of the root guard feature can cause a loss of connectivity.

Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is enabled on the entire switched network. Loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in PVST+ or rapid-PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in MST mode, BPDUs are not sent on nonboundary ports only if the interface is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the interface in all MST instances.

STP PortFast Port Types

You can configure a spanning tree port as an edge port, a network port, or a normal port. A port can be in only one of these states at a given time. The default spanning tree port type is normal. You can configure the port type either globally or per interface.

Depending on the type of device to which the interface is connected, you can configure a spanning tree port as one of these port types:

- A PortFast edge port—is connected to a Layer 2 host. This can be either an access port or an edge trunk port (**portfast edge trunk**). This type of port interface immediately transitions to the forwarding state, bypassing the listening and learning states. Use PortFast edge on Layer 2 access ports connected to a single workstation or server to allow those devices to connect to the network immediately, rather than waiting for spanning tree to converge.

Even if the interface receives a bridge protocol data unit (BPDU), spanning tree does not place the port into the blocking state. Spanning tree sets the port's operating state to *non-port fast* even if the configured state remains *port fast edge* and starts participating in the topology change.



Note If you configure a port connected to a Layer 2 switch or bridge as an edge port, you might create a bridging loop.

- A PortFast network port—is connected only to a Layer 2 switch or bridge. Bridge Assurance is enabled only on PortFast network ports. For more information, refer to *Bridge Assurance*.



Note If you configure a port that is connected to a Layer 2 host as a spanning tree network port, the port will automatically move into the blocking state.

- A PortFast normal port—is the default type of spanning tree port.



Note Beginning with Cisco IOS Release 15.2(4)E, or IOS XE 3.8.0E, if you enter the **spanning-tree portfast** [trunk] command in the global or interface configuration mode, the system automatically saves it as **spanning-tree portfast edge** [trunk].

Bridge Assurance

You can use Bridge Assurance to help prevent looping conditions that are caused by unidirectional links (one-way traffic on a link or port), or a malfunction in a neighboring switch. Here a malfunction refers to a switch that is not able to run STP any more, while still forwarding traffic (a brain dead switch).

BPDUs are sent out on all operational network ports, including alternate and backup ports, for each hello time period. Bridge Assurance monitors the receipt of BPDUs on point-to-point links on all network ports. When a port does not receive BPDUs within the allotted hello time period, the port is put into a blocked state (the same as a port inconsistent state, which stops forwarding of frames). When the port resumes receipt of BPDUs, the port resumes normal spanning tree operations.

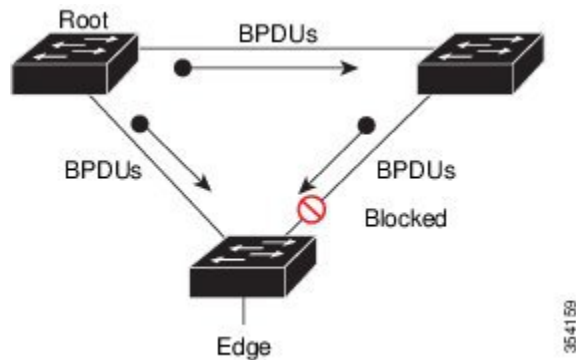


Note Only Rapid PVST+ and MST spanning tree protocols support Bridge Assurance. PVST+ does not support Bridge Assurance.

The following example shows how Bridge Assurance protects your network from bridging loops.

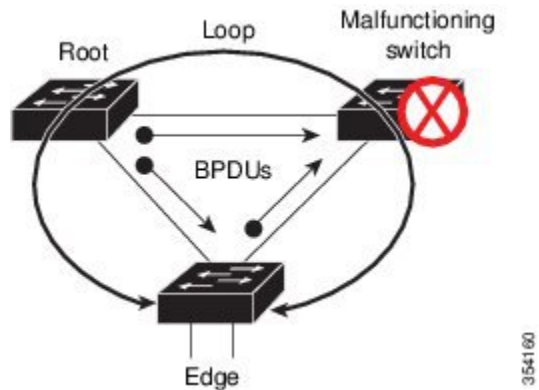
The following figure shows a network with normal STP topology.

Figure 41: Network with Normal STP Topology



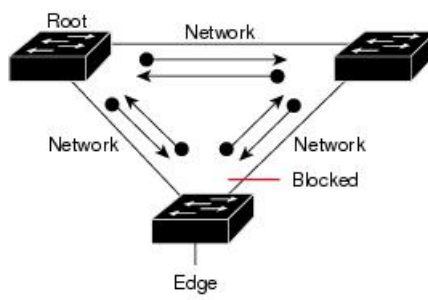
The following figure demonstrates a potential network problem when the device fails (brain dead) and Bridge Assurance is not enabled on the network.

Figure 42: Network Loop Due to a Malfunctioning Switch



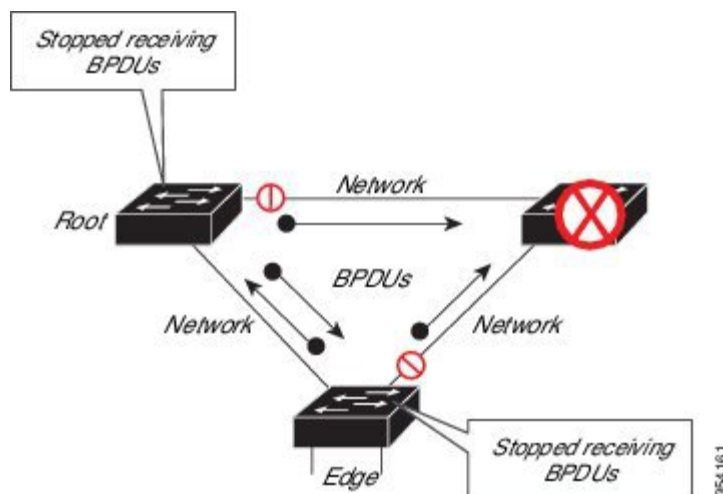
The following figure shows the network with Bridge Assurance enabled, and the STP topology progressing normally with bidirectional BDPUs issuing from every STP network port.

Figure 43: Network with STP Topology Running Bridge Assurance



The following figure shows how the potential network problem shown in figure *Network Loop Due to a Malfunctioning Switch* does not occur when you have Bridge Assurance enabled on your network.

Figure 44: Network Problem Averted with Bridge Assurance Enabled



The system generates syslog messages when a port is block and unblocked. The following sample output shows the log that is generated for each of these states:

BRIDGE_ASSURANCE_BLOCK

```
Sep 17 09:48:16.249 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_BLOCK: Bridge Assurance blocking port
GigabitEthernet1/0/1 on VLAN0001.
```

BRIDGE_ASSURANCE_UNBLOCK

```
Sep 17 09:48:58.426 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_UNBLOCK: Bridge Assurance unblocking
port GigabitEthernet1/0/1 on VLAN0001.
```

Follow these guidelines when enabling Bridge Assurance:

- It can only be enabled or disabled globally.
- It applies to all operational network ports, including alternate and backup ports.

- Only Rapid PVST+ and MST spanning tree protocols support Bridge Assurance. PVST+ does not support Bridge Assurance.
- For Bridge Assurance to work properly, it must be supported and configured on both ends of a point-to-point link. If the device on one side of the link has Bridge Assurance enabled and the device on the other side does not, the connecting port is blocked and in a Bridge Assurance inconsistent state. We recommend that you enable Bridge Assurance throughout your network.
- To enable Bridge Assurance on a port, BPDU filtering and BPDU Guard must be disabled.
- You can enable Bridge Assurance in conjunction with Loop Guard.
- You can enable Bridge Assurance in conjunction with Root Guard. The latter is designed to provide a way to enforce the root bridge placement in the network.

How to Configure Optional Spanning-Tree Features

Enabling PortFast

An interface with the PortFast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

If you enable the voice VLAN feature, the PortFast feature is automatically enabled. When you disable voice VLAN, the PortFast feature is not automatically disabled.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.



Caution Use PortFast only when connecting a single end station to an access or trunk port. Enabling this feature on an interface connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree portfast** [**trunk**]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2	Specifies an interface to configure, and enters interface configuration mode.
Step 4	spanning-tree portfast [trunk] Example: Switch(config-if)# spanning-tree portfast trunk	<p>Enables PortFast on an access port connected to a single workstation or server.</p> <p>By specifying the trunk keyword, you can enable PortFast on a trunk port.</p> <p>Note To enable PortFast on trunk ports, you must use the spanning-tree portfast trunk interface configuration command. The spanning-tree portfast command will not work on trunk ports.</p> <p>Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable PortFast on a trunk port.</p> <p>By default, PortFast is disabled on all interfaces.</p>
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

What to do next

You can use the **spanning-tree portfast default** global configuration command to globally enable the PortFast feature on all nontrunking ports.

Enabling BPDU Guard

You can enable the BPDU guard feature if your switch is running PVST+, Rapid PVST+, or MSTP.



Caution Configure PortFast edge only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast edge bpduguard default**
4. **interface *interface-id***
5. **spanning-tree portfast edge**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree portfast edge bpduguard default Example: <pre>Switch(config)# spanning-tree portfast edge bpduguard default</pre>	Globally enables BPDU guard.
Step 4	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/2</pre>	Specifies the interface connected to an end station, and enters interface configuration mode.
Step 5	spanning-tree portfast edge Example: <pre>Switch(config-if)# spanning-tree portfast edge</pre>	Enables the PortFast edge feature.
Step 6	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config-if)# end	

What to do next

To prevent the port from shutting down, you can use the **errdisable detect cause bpduguard shutdown vlan** global configuration command to shut down just the offending VLAN on the port where the violation occurred.

You also can use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any port without also enabling the PortFast edge feature. When the port receives a BPDU, it is put it in the error-disabled state.

Enabling BPDU Filtering

You can also use the **spanning-tree bpdudfilter enable** interface configuration command to enable BPDU filtering on any interface without also enabling the PortFast edge feature. This command prevents the interface from sending or receiving BPDUs.



Caution Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature if your switch is running PVST+, Rapid PVST+, or MSTP.



Caution Configure PortFast edge only on interfaces that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast edge bpdudfilter default**
4. **interface *interface-id***
5. **spanning-tree portfast edge**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree portfast edge bpdufilter default Example: <pre>Switch(config)# spanning-tree portfast edge bpdufilter default</pre>	Globally enables BPDU filtering. By default, BPDU filtering is disabled.
Step 4	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/2</pre>	Specifies the interface connected to an end station, and enters interface configuration mode.
Step 5	spanning-tree portfast edge Example: <pre>Switch(config-if)# spanning-tree portfast edge</pre>	Enables the PortFast edge feature on the specified interface.
Step 6	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Enabling UplinkFast for Use with Redundant Links



Note When you enable UplinkFast, it affects all VLANs on the switch or switch stack. You cannot configure UplinkFast on an individual VLAN.

You can configure the UplinkFast or the Cross-Stack UplinkFast (CSUF) feature for Rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

This procedure is optional. Follow these steps to enable UplinkFast and CSUF.

Before you begin

UplinkFast cannot be enabled on VLANs that have been configured with a switch priority. To enable UplinkFast on a VLAN with switch priority configured, first restore the switch priority on the VLAN to the default value using the **no spanning-tree vlan *vlan-id* priority** global configuration command.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree uplinkfast [max-update-rate <i>pkts-per-second</i>] Example: <pre>Switch(config)# spanning-tree uplinkfast max-update-rate 200</pre>	Enables UplinkFast. (Optional) For <i>pkts-per-second</i> , the range is 0 to 32000 packets per second; the default is 150. If you set the rate to 0, station-learning frames are not generated, and the spanning-tree topology converges more slowly after a loss of connectivity. When you enter this command, CSUF also is enabled on all nonstack port interfaces.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

When UplinkFast is enabled, the switch priority of all VLANs is set to 49152. If you change the path cost to a value less than 3000 and you enable UplinkFast or UplinkFast is already enabled, the path cost of all interfaces and VLAN trunks is increased by 3000 (if you change the path cost to 3000 or above, the path cost is not altered). The changes to the switch priority and the path cost reduce the chance that a switch will become the root switch.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

When you enable the UplinkFast feature using these instructions, CSUF is automatically globally enabled on nonstack port interfaces.

Disabling UplinkFast

This procedure is optional.

Follow these steps to disable UplinkFast and Cross-Stack UplinkFast (CSUF).

Before you begin

UplinkFast must be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree uplinkfast**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	no spanning-tree uplinkfast Example: Switch(config)# no spanning-tree uplinkfast	Disables UplinkFast and CSUF on the switch and all of its VLANs.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

When you disable the UplinkFast feature using these instructions, CSUF is automatically globally disabled on nonstack port interfaces.

Enabling BackboneFast

You can enable BackboneFast to detect indirect link failures and to start the spanning-tree reconfiguration sooner.

You can configure the BackboneFast feature for Rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

This procedure is optional. Follow these steps to enable BackboneFast on the switch.

Before you begin

If you use BackboneFast, you must enable it on all switches in the network. BackboneFast is not supported on Token Ring VLANs. This feature is supported for use with third-party switches.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree backbonefast**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree backbonefast Example: Switch(config)# spanning-tree backbonefast	Enables BackboneFast.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Enabling EtherChannel Guard

You can enable EtherChannel guard to detect an EtherChannel misconfiguration if your device is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable EtherChannel Guard on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree etherchannel guard misconfig**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree etherchannel guard misconfig Example: Switch(config)# spanning-tree etherchannel guard misconfig	Enables EtherChannel guard.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

What to do next

You can use the **show interfaces status err-disabled** privileged EXEC command to show which device ports are disabled because of an EtherChannel misconfiguration. On the remote device, you can enter the **show etherchannel summary** privileged EXEC command to verify the EtherChannel configuration.

After the configuration is corrected, enter the **shutdown** and **no shutdown** interface configuration commands on the port-channel interfaces that were misconfigured.

Enabling Root Guard

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. Do not enable the root guard on interfaces to be used by the UplinkFast feature. With UplinkFast, the backup interfaces (in the blocked state) replace the root port in the case of a failure. However, if root guard is also enabled, all the

backup interfaces used by the UplinkFast feature are placed in the root-inconsistent state (blocked) and are prevented from reaching the forwarding state.



Note You cannot enable both root guard and loop guard at the same time.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable root guard on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree guard root**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2	Specifies an interface to configure, and enters interface configuration mode.
Step 4	spanning-tree guard root Example: Switch(config-if)# spanning-tree guard root	Enables root guard on the interface. By default, root guard is disabled on all interfaces.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on interfaces that are considered point-to-point by the spanning tree.



Note You cannot enable both loop guard and root guard at the same time.

You can enable this feature if your device is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional. Follow these steps to enable loop guard on the device.

SUMMARY STEPS

1. Enter one of the following commands:
 - **show spanning-tree active**
 - **show spanning-tree mst**
2. **configure terminal**
3. **spanning-tree loopguard default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Enter one of the following commands: <ul style="list-style-type: none"> • show spanning-tree active • show spanning-tree mst Example: <pre>Switch# show spanning-tree active</pre> or <pre>Switch# show spanning-tree mst</pre>	Verifies which interfaces are alternate or root ports.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree loopguard default Example: <pre>Switch(config)# spanning-tree loopguard default</pre>	Enables loop guard. By default, loop guard is disabled.

	Command or Action	Purpose
Step 4	end Example: Switch(config) # end	Returns to privileged EXEC mode.

Enabling PortFast Port Types

This section describes the different steps to enable Portfast Port types.

Configuring the Default Port State Globally

To configure the default PortFast state, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast [edge | network | normal] default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree portfast [edge network normal] default Example: Switch(config) # spanning-tree portfast default	Configures the default state for all interfaces on the switch. You have these options: <ul style="list-style-type: none"> • (Optional) edge—Configures all interfaces as edge ports. This assumes all ports are connected to hosts/servers. • (Optional) network—Configures all interfaces as spanning tree network ports. This assumes all ports are connected to switches and bridges. Bridge Assurance is enabled on all network ports by default.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) normal—Configures all interfaces normal spanning tree ports. These ports can be connected to any type of device. • default—The default port type is normal.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring PortFast Edge on a Specified Interface

Interfaces configured as edge ports immediately transition to the forwarding state, without passing through the blocking or learning states, on linkup.



Note Because the purpose of this type of port is to minimize the time that access ports must wait for spanning tree to converge, it is most effective when used on access ports. If you enable PortFast edge on a port connecting to another switch, you risk creating a spanning tree loop.

To configure an edge port on a specified interface, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast edge** [trunk]
5. **end**
6. **show running interface** *interface-id* | **port-channel** *port_channel_number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> port-channel <i>port_channel_number</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1 port-channel port_channel_number</pre>	Specifies an interface to configure.
Step 4	spanning-tree portfast edge [trunk] Example: <pre>Switch(config-if)# spanning-tree portfast trunk</pre>	Enables edge behavior on a Layer 2 access port connected to an end workstation or server. <ul style="list-style-type: none"> • (Optional) trunk—Enables edge behavior on a trunk port. Use this keyword if the link is a trunk. Use this command only on ports that are connected to end host devices that terminate VLANs and from which the port should never receive STP BPDUs. Such end host devices include workstations, servers, and ports on routers that are not configured to support bridging. • Use the no version of the command to disable PortFast edge.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Exits configuration mode.
Step 6	show running interface <i>interface-id</i> port-channel <i>port_channel_number</i> Example: <pre>Switch# show running interface gigabitethernet 1/0/1 port-channel port_channel_number</pre>	Verifies the configuration.

Configuring a PortFast Network Port on a Specified Interface

Ports that are connected to Layer 2 switches and bridges can be configured as network ports.



Note Bridge Assurance is enabled only on PortFast network ports. For more information, refer to *Bridge Assurance*.

To configure a port as a network port, perform this task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast network**
5. **end**

6. `show running interface interface-id | port-channel port_channel_number`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface interface-id port-channel port_channel_number Example: <pre>Switch(config)# interface gigabitethernet 1/0/1 port-channel port_channel_number</pre>	Specifies an interface to configure.
Step 4	spanning-tree portfast network Example: <pre>Switch(config-if)# spanning-tree portfast network</pre>	Enables edge behavior on a Layer 2 access port connected to an end workstation or server. <ul style="list-style-type: none"> • Configures the port as a network port. If you have enabled Bridge Assurance globally, it automatically runs on a spanning tree network port. • Use the no version of the command to disable PortFast.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Exits configuration mode.
Step 6	show running interface interface-id port-channel port_channel_number Example: <pre>Switch# show running interface gigabitethernet 1/0/1 port-channel port_channel_number</pre>	Verifies the configuration.

Enabling Bridge Assurance

To configure the Bridge Assurance, perform the steps given below:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree bridge assurance**
4. **end**
5. **show spanning-tree summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	spanning-tree bridge assurance Example: Switch(config)# spanning-tree bridge assurance	Enables Bridge Assurance on all network ports on the switch. Bridge Assurance is enabled by default. Use the no version of the command to disable the feature. Disabling Bridge Assurance causes all configured network ports to behave as normal spanning tree ports.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show spanning-tree summary Example: Switch# show spanning-tree summary	Displays spanning tree information and shows if Bridge Assurance is enabled.

Examples

Examples: Configuring PortFast Edge on a Specified Interface

This example shows how to enable edge behavior on GigabitEthernet interface **1/0/1**:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# spanning-tree portfast edge
Switch(config-if)# end
Switch#
```

This example shows how to verify the configuration:

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
spanning-tree portfast edge
end
```

This example shows how you can display that port GigabitEthernet 1/0/1 is currently in the edge state:

```
Switch# show spanning-tree vlan 200
VLAN0200
Spanning tree enabled protocol rstp
Root ID Priority 2
Address 001b.2a68.5fc0
Cost 3
Port 125 (GigabitEthernet5/9)
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 2 (priority 0 sys-id-ext 2)
Address 7010.5c9c.5200
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 0 sec
Interface Role Sts Cost Prio.Nbr Type
-----
G1/0/1 Desg FWD 4 128.1 P2p Edge
```

Examples: Configuring a PortFast Network Port on a Specified Interface

This example shows how to configure GigabitEthernet interface 1/0/1 as a network port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# spanning-tree portfast network
Switch(config-if)# end
Switch#
```

This example shows how to verify the configuration:

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
spanning-tree portfast network
end
```

This example shows the output for show spanning-tree vlan

```

Switch# show spanning-tree vlan
Sep 17 09:51:36.370 PDT: %SYS-5-CONFIG_I: Configured from console by console2

VLAN0002
  Spanning tree enabled protocol rstp
  Root ID    Priority    2
            Address    7010.5c9c.5200
            This bridge is the root
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    2          (priority 0 sys-id-ext 2)
            Address    7010.5c9c.5200
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
            Aging Time 0      sec

Interface                Role Sts Cost          Prio.Nbr Type
-----
Gi1/0/1                  Desg FWD 4             128.1   P2p Edge
Po4                       Desg FWD 3             128.480 P2p Network
Gi4/0/1                  Desg FWD 4             128.169 P2p Edge
Gi4/0/47                 Desg FWD 4             128.215 P2p Network

Switch#

```

Example: Configuring Bridge Assurance

This output shows port GigabitEthernet 1/0/1 has been configured as a network port and it is currently in the Bridge Assurance inconsistent state.



Note The output shows the port type as network and *BA_Inc, indicating that the port is in an inconsistent state.

```

Switch# show spanning-tree
VLAN0010
  Spanning tree enabled protocol rstp
  Root ID Priority 32778
  Address 0002.172c.f400
  This bridge is the root
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
  Address 0002.172c.f400
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Aging Time 300
  Interface Role Sts Cost Prio. Nbr Type
  -----
Gi1/0/1    Desg BKN*4 128.270 Network, P2p *BA_Inc

```

The example shows the output for show spanning-tree summary.

```

Switch#sh spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001-VLAN0002, VLAN0128
EtherChannel misconfig guard          is enabled
Extended system ID                    is enabled
Portfast Default                      is network
Portfast Edge BPDU Guard Default      is disabled
Portfast Edge BPDU Filter Default     is disabled

```

```

Loopguard Default          is enabled
PVST Simulation Default    is enabled but inactive in rapid-pvst mode
Bridge Assurance           is enabled
UplinkFast                 is disabled
BackboneFast               is disabled
Configured Pathcost method used is short

```

Name	Blocking	Listening	Learning	Forwarding	STP Active
VLAN0001	0	0	0	5	5
VLAN0002	0	0	0	4	4
VLAN0128	0	0	0	4	4
3 vlans	0	0	0	13	13

```
Switch#
```

Monitoring the Spanning-Tree Status

Table 49: Commands for Monitoring the Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface <i>interface-id</i>	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface <i>interface-id</i>	Displays MST information for the specified interface.
show spanning-tree summary [totals]	Displays a summary of interface states or displays the total line spanning-tree state section.
show spanning-tree mst interface <i>interface-id</i> portfast edge	Displays spanning-tree portfast information for the specified interface.



CHAPTER 27

Configuring Bidirection Forwarding Detection

- [Finding Feature Information, on page 513](#)
- [Prerequisites for Bidirectional Forwarding Detection, on page 513](#)
- [Restrictions for Bidirectional Forwarding Detection, on page 513](#)
- [Information About Bidirectional Forwarding Detection, on page 514](#)
- [How to Configure Bidirectional Forwarding Detection, on page 518](#)
- [Configuration Examples for Bidirectional Forwarding Detection, on page 530](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Bidirectional Forwarding Detection

Prerequisites for BFD include:

- The switch's feature set is IP Base or higher. The IP Base feature set supports only Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing, without BFD. The IP service feature set supports EIGRP with BFD.
- IP routing must be enabled on all participating switches
- Before BFD is deployed, configure one of the IP routing protocols supported by BFD on the switches. Also, implement fast convergence for the routing protocol that you plan to use.

Restrictions for Bidirectional Forwarding Detection

Restrictions for BFD include:

- BFD works only for directly connected neighbors. BFD neighbors must be no more than one IP hop away. Multihop configurations are not supported.
- The switch supports up to 100 BFD sessions with a minimum hello interval of 100 ms and a multiplier of 3. The multiplier specifies the minimum number of consecutive packets that can be missed before a session is declared down.
- To enable echo mode the peer system must be configured with the no ip redirects command.

Information About Bidirectional Forwarding Detection

BFD Operation

BFD provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, and forwarding planes.

BFD is a detection protocol that you enable at the interface and routing protocol levels. Cisco supports the BFD asynchronous mode, which depends on the sending of BFD control packets between two systems to activate and maintain BFD neighbor sessions between routers. Therefore, in order for a BFD session to be created, you must configure BFD on both systems (or BFD peers). Once BFD has been enabled on the interfaces and at the router level for the appropriate routing protocols, a BFD session is created, BFD timers are negotiated, and the BFD peers will begin to send BFD control packets to each other at the negotiated interval.

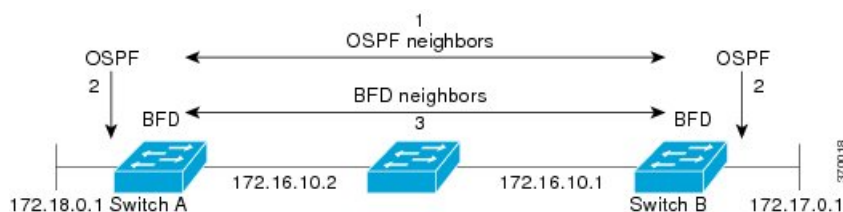
Cisco supports BFD echo mode. Echo packets are sent by the forwarding engine and are forwarded back along the same path to perform detection. The BFD session at the other end does not participate in the actual forwarding of the echo packets.

This section includes the following subsections:

Neighbor Relationships

BFD provides fast BFD peer failure detection times independently of all media types, encapsulations, topologies, and routing protocols BGP, EIGRP, IS-IS, and OSPF. By sending rapid failure detection notices to the routing protocols in the local router to initiate the routing table recalculation process, BFD contributes to greatly reduced overall network convergence time. The figure below shows a simple network with two routers running OSPF and BFD. When OSPF discovers a neighbor (1) it sends a request to the local BFD process to initiate a BFD neighbor session with the OSPF neighbor router (2). The BFD neighbor session with the OSPF neighbor router is established (3).

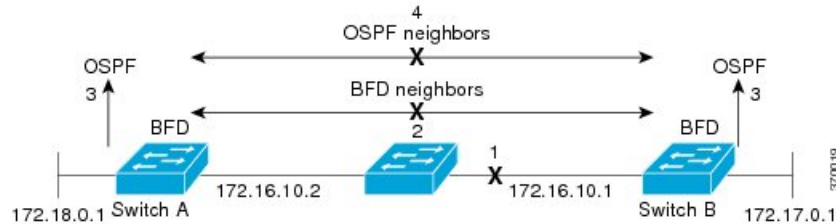
Figure 45: Establishing a BFD Neighbor Relationship



The figure below shows what happens when a failure occurs in the network (1). The BFD neighbor session with the OSPF neighbor router is torn down (2). BFD notifies the local OSPF process that the BFD neighbor

is no longer reachable (3). The local OSPF process tears down the OSPF neighbor relationship (4). If an alternative path is available, the routers will immediately start converging on it.

Figure 46: Tearing Down an OSPF Neighbor Relationship



A routing protocol needs to register with BFD for every neighbor it acquires. Once a neighbor is registered, BFD initiates a session with the neighbor if a session does not already exist.

OSPF registers with BFD when:

- A neighbor finite state machine (FSM) transitions to full state.
- Both OSPF BFD and BFD are enabled.

On broadcast interfaces, OSPF establishes a BFD session only with the designated router (DR) and backup designated router (BDR), but not between any two routers in DROTHER state.

BFD Detection of Failures

Once a BFD session has been established and timer negotiations are complete, BFD peers send BFD control packets that act in the same manner as an IGP hello protocol to detect liveness, except at a more accelerated rate. The following information should be noted:

- BFD is a forwarding path failure detection protocol. BFD detects a failure, but the routing protocol must take action to bypass a failed peer.
 - Typically, BFD can be used at any protocol layer. However, the Cisco implementation of BFD supports only Layer 3 clients, in particular, the BGP, EIGRP, and OSPF routing protocol, and static routing.
- Cisco devices will use one BFD session for multiple client protocols in the Cisco implementation of BFD. For example, if a network is running OSPF and EIGRP across the same link to the same peer, only one BFD session will be established, and BFD will share session information with both routing protocols. However, IPv4 and IPv6 clients cannot share a BFD session.

BFD Version Interoperability

The switch supports BFD Version 1 as well as BFD Version 0. All BFD sessions come up as Version 1 by default and will be interoperable with Version 0. The system automatically performs BFD version detection, and BFD sessions between neighbors will run in the highest common BFD version between neighbors. For example, if one BFD neighbor is running BFD Version 0 and the other BFD neighbor is running Version 1, the session will run BFD Version 0. The output from the **show bfd neighbors [details]** command will verify which BFD version a BFD neighbor is running.

BFD Session Limits

The minimum number of BFD sessions that can be created varies with the “hello” interval. With “hello” intervals of 100ms, 100 sessions are permitted. More sessions are permitted at larger hello intervals. For a VLAN interface, the minimum “hello” interval is 600ms.

BFD Support for Nonbroadcast Media Interfaces

The BFD feature is supported on VLAN interfaces on the switch.

The **bfd interval** command must be configured on the interface to initiate BFD monitoring.

BFD Support for Nonstop Forwarding with Stateful Switchover

Typically, when a networking device restarts, all routing peers of that device detect that the device went down and then came back up. This transition results in a routing flap, which could spread across multiple routing domains. Routing flaps caused by routing restarts create routing instabilities, which are detrimental to the overall network performance. Nonstop forwarding (NSF) helps to suppress routing flaps in devices that are enabled with stateful switchover (SSO), thereby reducing network instability.

NSF allows for the forwarding of data packets to continue along known routes while the routing protocol information is being restored after a switchover. With NSF, peer networking devices do not experience routing flaps. Data traffic is forwarded through intelligent line cards or dual forwarding processors while the standby RP assumes control from the failed active RP during a switchover. The ability of line cards and forwarding processors to remain up through a switchover and to be kept current with the Forwarding Information Base (FIB) on the active RP is key to NSF operation.

In devices that support dual RPs, SSO establishes one of the RPs as the active processor; the other RP is designated as the standby processor, and then synchronizes information between them. A switchover from the active to the standby processor occurs when the active RP fails, when it is removed from the networking device, or when it is manually taken down for maintenance.

BFD Support for Stateful Switchover

The BFD protocol provides short-duration detection of failures in the path between adjacent forwarding engines. In network deployments that use dual RP switches (to provide redundancy), the switches have a graceful restart mechanism that protects the forwarding state during a switchover between the active RP and the standby RP.

Stateful BFD on the Standby RP

To ensure a successful switchover to the standby RP, the BFD protocol uses checkpoint messages to send session information from the active RP Cisco IOS instance to the standby RP Cisco IOS instance. The session information includes local and remote discriminators, adjacent router timer information, BFD setup information, and session-specific information such as the type of session and the session version. In addition, the BFD protocol sends session creation and deletion checkpoint messages to create or delete a session on the standby RP.

The BFD sessions on the standby RP do not receive or send packets and do not process expired timers. These sessions wait for a switchover to occur and then send packets for any active sessions so that sessions do not time out on adjacent switches.

When the BFD protocol on the standby RP is notified of a switchover it changes its state to active, registers itself with Cisco Express Forwarding so that it can receive packets, and then sends packets for any elements that have expired.

BFD also uses checkpoint messages to ensure that sessions created by clients on the active RP are maintained during a switchover. When a switchover occurs, BFD starts an SSO reclaim timer. Clients must reclaim their sessions within the duration specified by the reclaim timer or else the session is deleted.

Timer values are different based on the number of BFD sessions and the platform.

Table 50: BFD Timer Values on the switch

Maximum Number of BFD Sessions	BFD Session Type	Minimum Timer Value (ms)	Clients	Comments
100	Async/echo	100 multiplier 3	All	A multiple of 5 is recommended for SSO switches.

BFD Support for Static Routing

Unlike dynamic routing protocols, such as OSPF and BGP, static routing has no method of peer discovery. Therefore, when BFD is configured, the reachability of the gateway is completely dependent on the state of the BFD session to the specified neighbor. Unless the BFD session is up, the gateway for the static route is considered unreachable, and therefore the affected routes will not be installed in the appropriate Routing Information Base (RIB).

For a BFD session to be successfully established, BFD must be configured on the interface on the peer and there must be a BFD client registered on the peer for the address of the BFD neighbor. When an interface is used by dynamic routing protocols, the latter requirement is usually met by configuring the routing protocol instances on each neighbor for BFD. When an interface is used exclusively for static routing, this requirement must be met by configuring static routes on the peers.

If a BFD configuration is removed from the remote peer while the BFD session is in the up state, the updated state of the BFD session is not signaled to the static static. This will cause the static route to remain in the RIB. The only workaround is to remove the IPv4 static BFD neighbor configuration so that the static route no longer tracks BFD session state.

Benefits of Using BFD for Failure Detection

When you deploy any feature, it is important to consider all the alternatives and be aware of any trade-offs being made.

The closest alternative to BFD in conventional EIGRP, BGP, and OSPF deployments is the use of modified failure detection mechanisms for EIGRP, BGP, and OSPF routing protocols.

If you set EIGRP hello and hold timers to their absolute minimums, the failure detection rate for EIGRP falls to within a one- to two-second range.

If you use fast hellos for either BGP or OSPF, these Interior Gateway Protocol (IGP) protocols reduce their failure detection mechanisms to a minimum of one second.

There are several advantages to implementing BFD over reduced timer mechanisms for routing protocols:

- Although reducing the EIGRP, BGP, and OSPF timers can result in minimum detection timer of one to two seconds, BFD can provide failure detection in less than one second.
- Because BFD is not tied to any particular routing protocol, it can be used as a generic and consistent failure detection mechanism for EIGRP, BGP, and OSPF.

- Because some parts of BFD can be distributed to the data plane, it can be less CPU-intensive than the reduced EIGRP, BGP, and OSPF timers, which exist wholly at the control plane.

How to Configure Bidirectional Forwarding Detection

You start a BFD process by configuring BFD on the interface. When the BFD process is started, no entries are created in the adjacency database; in other words, no BFD control packets are sent or received. BFD echo mode, which is supported in BFD Version 1.

BFD echo packets are sent and received, in addition to BFD control packets. The adjacency creation takes places once you have configured BFD support for the applicable routing protocols. This section contains the following procedures:

Configuring BFD Session Parameters on the Interface

Perform this task to configure BFD on an interface by setting the baseline BFD session parameters on the interface. Repeat this task on each interface over which you want to run BFD sessions to BFD neighbors.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **bfd interval** *milliseconds min_rx milliseconds multiplier interval-multiplier*
5. **no bfd echo**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface GigabitEthernet 6/1	Specifies an interface type and number, and places the device in interface configuration mode.
Step 4	bfd interval <i>milliseconds min_rx milliseconds multiplier interval-multiplier</i>	Enables BFD on the interface. Disables BFD echo mode to enable Hardware Off-load.

	Command or Action	Purpose
	Example: <pre>Switch(config-if)# no bfd echo</pre>	
Step 5	no bfd echo Example: <pre>Switch(config-if)# no bfd echo</pre>	Disables BFD echo mode to enable Hardware Off-load.
Step 6	end Example: <pre>Switch(config-if)# end</pre>	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring BFD Support for Dynamic Routing Protocols

You can enable BFD support for dynamic routing protocols at the device level to enable BFD support globally for all interfaces or you can configure BFD on a per-interface basis at the interface level.

This section describes the following tasks:

Configuring BFD Support for BGP

Perform this task to configure BFD support for Border Gateway Protocol (BGP) so that BGP is a registered protocol with BFD and will receive forwarding path detection failure messages from BFD.

Before you begin

BGP must be running on all participating switches.

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. See the Configuring BFD Session Parameters on the Interface section for more information.



Note Output from the **show bfd neighbors details** command shows the configured intervals. The output does not show intervals that were changed because hardware-offloaded BFD sessions were configured with Tx and Rx intervals that are not multiples of 50 ms.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp *as-tag***
4. **neighbor *ip-address* fall-over bfd**
5. **end**
6. **show bfd neighbors [details]**
7. **show ip bgp neighbor**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	router bgp <i>as-tag</i> Example: Switch(config)# router bgp tag1	Specifies a BGP process and enters router configuration mode.
Step 4	neighbor <i>ip-address</i> fall-over bfd Example: Switch(config-router)# neighbor 172.16.10.2 fall-over bfd	Enables BFD support for fallover.
Step 5	end Example: Switch(config-router)# end	Exits router configuration mode and returns to privileged EXEC mode.
Step 6	show bfd neighbors [details] Example: Switch# show bfd neighbors detail	(Optional) Verifies that the BFD neighbor is active and displays the routing protocols that BFD has registered.
Step 7	show ip bgp neighbor Example: Switch# show ip bgp neighbor	(Optional) Displays information about BGP and TCP connections to neighbors.

Configuring BFD Support for EIGRP

This section describes the procedure for configuring BFD support for EIGRP so that EIGRP is a registered protocol with BFD and will receive forwarding path detection failure messages from BFD. There are two methods for enabling BFD support for EIGRP:

- You can enable BFD for all of the interfaces for which EIGRP is routing by using the **bfd all-interfaces** command in router configuration mode.
- You can enable BFD for a subset of the interfaces for which EIGRP is routing by using the **bfd interface *type number*** command in router configuration mode.

Before you begin

EIGRP must be running on all participating switches.

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. For more information, see the "Configuring BFD Session Parameters on the Interface".

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp *as-number***
4. Do one of the following:
 - **bfd all-interfaces**
 - **bfd interface *type number***
5. **end**
6. **show bfd neighbors [details]**
7. **show ip eigrp interfaces [*type number*] [*as-number*] [detail]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	router eigrp <i>as-number</i> Example: Switch(config)# router eigrp 123	Configures the EIGRP routing process and enters router configuration mode.
Step 4	Do one of the following: <ul style="list-style-type: none"> • bfd all-interfaces • bfd interface <i>type number</i> Example: Switch(config-router)# bfd all-interfaces Example: Switch(config-router)# bfd interface FastEthernet 6/1	Enables BFD globally on all interfaces associated with the EIGRP routing process. or Enables BFD on a per-interface basis for one or more interfaces associated with the EIGRP routing process.

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config-router) end</pre>	Exits router configuration mode and returns to privileged EXEC mode.
Step 6	show bfd neighbors [details] Example: <pre>Switch# show bfd neighbors details</pre>	(Optional) Verifies that the BFD neighbor is active and displays the routing protocols that BFD has registered.
Step 7	show ip eigrp interfaces [type number] [as-number] [detail] Example: <pre>Switch# show ip eigrp interfaces detail</pre>	(Optional) Displays the interfaces for which BFD support for EIGRP has been enabled.

Configuring BFD Support for OSPF

This section describes the procedures for configuring BFD support for OSPF so that OSPF is a registered protocol with BFD and will receive forwarding path detection failure messages from BFD. You can either configure BFD support for OSPF globally on all interfaces or configure it selectively on one or more interfaces.

There are two methods for enabling BFD support for OSPF:

- You can enable BFD for all of the interfaces for which OSPF is routing by using the **bfd all-interfaces** command in router configuration mode. You can disable BFD support on individual interfaces using the **ip ospf bfd [disable]** command in interface configuration mode.
- You can enable BFD for a subset of the interfaces for which OSPF is routing by using the **ip ospf bfd** command in interface configuration mode.

See the following sections for tasks for configuring BFD support for OSPF:

Configuring BFD Support for OSPF for All Interfaces

Perform this task to configure BFD for all OSPF interfaces.

If you do not want to configure BFD on all OSPF interfaces and would rather configure BFD support specifically for one or more interfaces, see the [Configuring OSPF Support for BFD over IPv4 for One or More Interfaces](#) section.

Before you begin

Open Shortest Path First (OSPF) must be running on all participating switches.

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. For more information, see the “[Configuring BFD Session Parameters on the Interface](#)” section.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **switch ospf *process-id***
4. **bfd all-interfaces**
5. **end**
6. **show bfd neighbors [details]**
7. **show ip ospf**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	switch ospf <i>process-id</i> Example: Switch(config)# router ospf 4	Specifies an OSPF process and enters router configuration mode.
Step 4	bfd all-interfaces Example: Switch(config-router)# bfd all-interfaces	Enables BFD globally on all interfaces associated with the OSPF routing process.
Step 5	end Example: Switch(config-router)# end	Exits router configuration mode and returns to privileged EXEC mode.
Step 6	show bfd neighbors [details] Example: Switch# show bfd neighbors detail	(Optional) Displays information that can help verify if the BFD neighbor is active and displays the routing protocols that BFD has registered.
Step 7	show ip ospf Example: Switch# show ip ospf	(Optional) Displays information that can help verify if BFD for OSPF has been enabled.

Configuring BFD Support for OSPF for One or More Interfaces

Perform this task to configure BFD for all OSPF interfaces.

If you do not want to configure BFD on all OSPF interfaces and would rather configure BFD support specifically for one or more interfaces, see the Configuring OSPF Support for BFD over IPv4 for One or More Interfaces section.

Before you begin

OSPF must be running on all participating switches.

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. For more information, see the “Configuring BFD Session Parameters on the Interface” section.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip ospf bfd** [**disable**]
5. **end**
6. **show bfd neighbors** [**details**]
7. **show ip ospf**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface fastethernet 6/1	(Optional) Enters interface configuration mode.
Step 4	ip ospf bfd [disable] Example: Switch(config-if)# ip ospf bfd	(Optional) Enables or disables BFD on a per-interface basis for one or more interfaces associated with the OSPF routing process. Note Use the disable keyword only if you enabled BFD on all of the interfaces that OSPF is associated with using the bfd all-interfaces command in router configuration mode.

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Exits interface configuration mode and returns to privileged EXEC mode.
Step 6	show bfd neighbors [details] Example: <pre>Switch# show bfd neighbors detail</pre>	(Optional) Displays information that can help verify if the BFD neighbor is active and displays the routing protocols that BFD has registered.
Step 7	show ip ospf Example: <pre>Switch# show ip ospf</pre>	(Optional) Displays information that can help verify if BFD for OSPF has been enabled.

Configuring BFD Support for Static Routing

Perform this task to configure BFD support for static routing. Repeat the steps in this procedure on each BFD neighbor. For more information, see the “Example: Configuring BFD Support for Static Routing” section

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no switchport**
5. **ip address** *ip-address mask*
6. **bfd interval** *milliseconds min_rx milliseconds multiplier interval-multiplier*
7. **exit**
8. **ip route static bfd** *interface-type interface-number ip-address* [**group** *group-name* [**passive**]]
9. **ip route** [**vrf** *vrf-name*] *prefix mask {ip-address | interface-type interface-number [ip-address]}* [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*]
10. **exit**
11. **show ip static route**
12. **show ip static route bfd**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface type number Example: Switch(config)# interface gigabitethernet 6/1	Configures an interface and enters interface configuration mode.
Step 4	no switchport Example: Switch(config)# no switchport	Changes the interface to Layer 3.
Step 5	ip address ip-address mask Example: Switch(config-if)# ip address 10.201.201.1 255.255.255.0	Configures an IP address for the interface.
Step 6	bfd interval milliseconds min_rx milliseconds multiplier interval-multiplier Example: Switch(config-if)# bfd interval 500 min_rx 500 multiplier 5	Enables BFD on the interface.
Step 7	exit Example: Switch(config-if)# exit	Exits interface configuration mode and returns to global configuration mode.
Step 8	ip route static bfd interface-type interface-number ip-address [group group-name [passive]] Example: Switch(config)# ip route static bfd serial 2/0 10.1.1.1 group group1 passive	Specifies a static route BFD neighbor. • The <i>interface-type</i> , <i>interface-number</i> , and <i>ip-address</i> arguments are required because BFD support exists only for directly connected neighbors.
Step 9	ip route [vrf vrf-name] prefix mask {ip-address interface-type interface-number [ip-address]} [dhcp] [distance] [name next-hop-name] [permanent track number] [tag tag] Example: Switch(config)# ip route 10.0.0.0 255.0.0.0 Gigabitethernet 6/1 10.201.201.2	Specifies a static route BFD neighbor.

	Command or Action	Purpose
Step 10	exit Example: <pre>Switch(config)# exit</pre>	Exits global configuration mode and returns to privileged EXEC mode.
Step 11	show ip static route Example: <pre>Switch# show ip static route</pre>	(Optional) Displays static route database information.
Step 12	show ip static route bfd Example: <pre>Switch# show ip static route bfd</pre>	(Optional) Displays information about the static BFD configuration from the configured BFD groups and non-group entries.

Configuring BFD Echo Mode

BFD echo mode is enabled by default, but you can disable it such that it can run independently in each direction.

BFD echo mode works with asynchronous BFD. Echo packets are sent by the forwarding engine and forwarded back along the same path in order to perform detection--the BFD session at the other end does not participate in the actual forwarding of the echo packets. The echo function and the forwarding engine are responsible for the detection process; therefore, the number of BFD control packets that are sent out between two BFD neighbors is reduced. In addition, because the forwarding engine is testing the forwarding path on the remote (neighbor) system without involving the remote system, there is an opportunity to improve the interpacket delay variance, thereby achieving quicker failure detection times than when using BFD Version 0 with BFD control packets for the BFD session.

Echo mode is described as without asymmetry when it is running on both sides (both BFD neighbors are running echo mode).

Prerequisites

BFD must be running on all participating switches.

Before using BFD echo mode, you must disable the sending of Internet Control Message Protocol (ICMP) redirect messages by entering the **no ip redirects** command, in order to avoid high CPU utilization.

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. See the Configuring BFD Session Parameters on the Interface section for more information.

Restrictions

BFD echo mode, which is supported in BFD Version 1.



Note BFD echo mode does not work in conjunction with Unicast Reverse Path Forwarding (uRPF) configuration. If BFD echo mode and uRPF configurations are enabled, then the sessions will flap.

Configuring the BFD Slow Timer

This task shows how to change the value of the BFD slow timer. Repeat the steps in this task for each BFD switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **bfd slow-timer** *milliseconds*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	bfd slow-timer <i>milliseconds</i> Example: Switch(config)# bfd slow-timer 12000	Configures the BFD slow timer.
Step 4	end Example: Switch(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Disabling BFD Echo Mode Without Asymmetry

This task shows how to disable BFD echo mode without asymmetry—no echo packets will be sent by the switch, and the switch will not forward BFD echo packets that are received from any neighbor switches.

Repeat the steps in this task for each BFD switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no bfd echo**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	no bfd echo Example: Switch(config)# no bfd echo	Disables BFD echo mode. <ul style="list-style-type: none"> • Use the no form to disable BFD echo mode.
Step 4	end Example: Switch(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Monitoring and Troubleshooting BFD

This section describes how to retrieve BFD information for maintenance and troubleshooting. The commands in these tasks can be entered as needed, in any order.

To monitor and troubleshoot BFD, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **show bfd neighbors [details]**
3. **debug bfd [packet | event]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show bfd neighbors [details] Example: Switch# show bfd neighbors details	(Optional) Displays the BFD adjacency database. <ul style="list-style-type: none"> • The details keyword shows all BFD protocol parameters and timers per neighbor.

	Command or Action	Purpose
Step 3	debug bfd [packet event] Example: Switch# debug bfd packet	(Optional) Displays debugging information about BFD packets.

Configuration Examples for Bidirectional Forwarding Detection

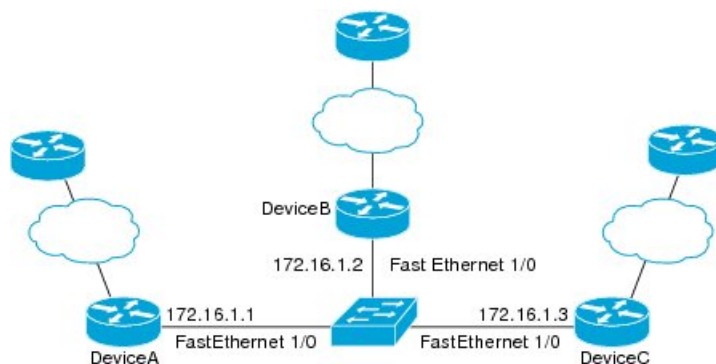
This section provides the following configuration examples:

Example: Configuring BFD in an EIGRP Network with Echo Mode Enabled by Default

In the following example, the EIGRP network contains DeviceA, DeviceB, and DeviceC. Fast Ethernet interface 1/0 on DeviceA is connected to the same network as Fast Ethernet interface 1/0 on Device B. Fast Ethernet interface 1/0 on DeviceB is connected to the same network as Fast Ethernet interface 1/0 on DeviceC.

DeviceA and DeviceB are running BFD Version 1, which supports echo mode, and DeviceC is running BFD Version 0, which does not support echo mode. The BFD sessions between DeviceC and its BFD neighbors are said to be running echo mode with asymmetry because echo mode will run on the forwarding path for DeviceA and DeviceB, and their echo packets will return along the same path for BFD sessions and failure detections, while their BFD neighbor DeviceC runs BFD Version 0 and uses BFD controls packets for BFD sessions and failure detections.

The figure below shows a large EIGRP network with several devices, three of which are BFD neighbors that are running EIGRP as their routing protocol.



The example, starting in global configuration mode, shows the configuration of BFD.

Configuration for DeviceA

```
interface Fast Ethernet0/0
no shutdown
ip address 10.4.9.14 255.255.255.0
duplex auto
speed auto
!
```

```
interface Fast Ethernet1/0
ip address 172.16.1.1 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
no shutdown
duplex auto
speed auto
!
router eigrp 11
network 172.16.0.0
bfd all-interfaces
auto-summary
!
ip default-gateway 10.4.9.1
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 10.4.9.1
ip route 172.16.1.129 255.255.255.255 10.4.9.1
!
no ip http server
!
logging alarm informational
!
control-plane
!
line con 0
exec-timeout 30 0
stopbits 1
line aux 0
stopbits 1
line vty 0 4
login
!
!
end
```

Configuration for DeviceB

```
!
interface Fast Ethernet0/0
no shutdown
ip address 10.4.9.34 255.255.255.0
duplex auto
speed auto
!
interface Fast Ethernet1/0
ip address 172.16.1.2 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
no shtdown
duplex auto
speed auto
!
router eigrp 11
network 172.16.0.0
bfd all-interfaces
auto-summary
!
ip default-gateway 10.4.9.1
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 10.4.9.1
ip route 172.16.1.129 255.255.255.255 10.4.9.1
!
no ip http server
!
logging alarm informational
```

```

!
control-plane
!
line con 0
exec-timeout 30 0
stopbits 1
line aux 0
stopbits 1
line vty 0 4
login
!
!
end

```

Configuration for DeviceC

```

!
!
interface Fast Ethernet0/0
no shutdown
ip address 10.4.9.34 255.255.255.0
duplex auto
speed auto
!
interface Fast Ethernet1/0
ip address 172.16.1.2 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
no shutdown
duplex auto
speed auto
!
router eigrp 11
network 172.16.0.0
bfd all-interfaces
auto-summary
!
ip default-gateway 10.4.9.1
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 10.4.9.1
ip route 172.16.1.129 255.255.255.255 10.4.9.1
!
no ip http server
!
logging alarm informational
!
control-plane
!
line con 0
exec-timeout 30 0
stopbits 1
line aux 0
stopbits 1
line vty 0 4
login
!
!
end

```

The output from the **show bfd neighbors details** command from DeviceA verifies that BFD sessions are created among all three devices and that EIGRP is registered for BFD support. The first group of output shows that DeviceC with the IP address 172.16.1.3 runs BFD Version 0 and therefore does not use the echo mode. The second group of output shows that DeviceB with the IP address 172.16.1.2 runs BFD Version 1, and the

50 millisecond BFD interval parameter had been adopted. The relevant command output is shown in bold in the output.

```
DeviceA# show bfd neighbors details
```

```
OurAddr
```

```
      NeighAddr
      LD/RD  RH/RS  Holdown(mult)  State  Int
172.16.1.1  172.16.1.3
      5/3    1(RH)   150 (3 )         Up    Fal/0
```

```
Session state is UP and not using echo function.
```

```
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 150(0), Hello (hits): 50(1364284)
Rx Count: 1351813, Rx Interval (ms) min/max/avg: 28/64/49 last: 4 ms ago
Tx Count: 1364289, Tx Interval (ms) min/max/avg: 40/68/49 last: 32 ms ago
Registered protocols: EIGRP
Uptime: 18:42:45
```

```
Last packet: Version: 0
```

```
- Diagnostic: 0
  I Hear You bit: 1      - Demand bit: 0
  Poll bit: 0           - Final bit: 0
  Multiplier: 3         - Length: 24
  My Discr.: 3          - Your Discr.: 5
  Min tx interval: 50000 - Min rx interval: 50000
  Min Echo interval: 0
```

```
OurAddr
```

```
      NeighAddr
      LD/RD  RH/RS  Holdown(mult)  State  Int
172.16.1.1  172.16.1.2
      6/1    Up      0 (3 )         Up    Fal/0
```

```
Session state is UP and using echo function with 50 ms interval.
```

```
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holdown (hits): 3000(0), Hello (hits): 1000(317)
Rx Count: 305, Rx Interval (ms) min/max/avg: 1/1016/887 last: 448 ms ago
Tx Count: 319, Tx Interval (ms) min/max/avg: 1/1008/880 last: 532 ms ago
Registered protocols: EIGRP
Uptime: 00:04:30
```

```
Last packet: Version: 1
```

```
- Diagnostic: 0
  State bit: Up          - Demand bit: 0
  Poll bit: 0           - Final bit: 0
  Multiplier: 3         - Length: 24
  My Discr.: 1          - Your Discr.: 6
  Min tx interval: 1000000 - Min rx interval: 1000000
  Min Echo interval: 50000
```

The output from the **show bfd neighbors details** command on Device B verifies that BFD sessions have been created and that EIGRP is registered for BFD support. As previously noted, DeviceA runs BFD Version 1, therefore echo mode is running, and DeviceC runs BFD Version 0, so echo mode does not run. The relevant command output is shown in bold in the output.

```
DeviceB# show bfd neighbors details
```

```
OurAddr      NeighAddr
```

Example: Configuring BFD in an EIGRP Network with Echo Mode Enabled by Default

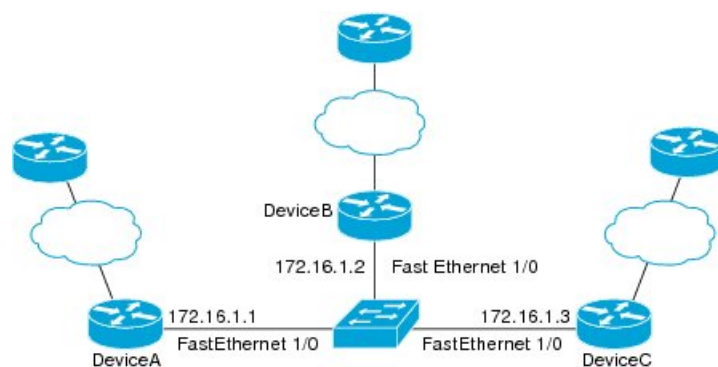
```

      LD/RD  RH/RS  Holdown(mult)  State  Int
172.16.1.2  172.16.1.1
      1/6    Up      0 (3)    Up      Fa1/0
Session state is UP and using echo function with 50 ms interval.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holdown (hits): 3000(0), Hello (hits): 1000(337)
Rx Count: 341, Rx Interval (ms) min/max/avg: 1/1008/882 last: 364 ms ago
Tx Count: 339, Tx Interval (ms) min/max/avg: 1/1016/886 last: 632 ms ago
Registered protocols: EIGRP
Uptime: 00:05:00
Last packet: Version: 1
    - Diagnostic: 0
      State bit: Up           - Demand bit: 0
      Poll bit: 0            - Final bit: 0
      Multiplier: 3          - Length: 24
      My Discr.: 6           - Your Discr.: 1
      Min tx interval: 1000000 - Min rx interval: 1000000
      Min Echo interval: 50000
OurAddr      NeighAddr

      LD/RD  RH/RS  Holdown(mult)  State  Int
172.16.1.2  172.16.1.3
      3/6    1(RH)  118 (3)    Up      Fa1/0
Session state is UP and not using echo function.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 150(0), Hello (hits): 50(5735)
Rx Count: 5731, Rx Interval (ms) min/max/avg: 32/72/49 last: 32 ms ago
Tx Count: 5740, Tx Interval (ms) min/max/avg: 40/64/50 last: 44 ms ago
Registered protocols: EIGRP
Uptime: 00:04:45
Last packet: Version: 0
    - Diagnostic: 0
      I Hear You bit: 1       - Demand bit: 0
      Poll bit: 0            - Final bit: 0
      Multiplier: 3          - Length: 24
      My Discr.: 6           - Your Discr.: 3
      Min tx interval: 50000  - Min rx interval: 50000
      Min Echo interval: 0

```

The figure below shows that Fast Ethernet interface 1/0 on DeviceB has failed. When Fast Ethernet interface 1/0 on DeviceB is shut down, the BFD statistics of the corresponding BFD sessions on DeviceA and DeviceC are reduced.



204950

When Fast Ethernet interface 1/0 on DeviceB fails, BFD will no longer detect Device B as a BFD neighbor for DeviceA or for DeviceC. In this example, Fast Ethernet interface 1/0 has been administratively shut down on DeviceB.

The following output from the **show bfd neighbors** command on DeviceA now shows only one BFD neighbor for DeviceA in the EIGRP network. The relevant command output is shown in bold in the output.

```
DeviceA# show bfd neighbors
OurAddr      NeighAddr

      LD/RD  RH/RS  Holdown(mult)  State      Int
172.16.1.1  172.16.1.3

      5/3    1(RH)   134 (3 )    Up        Fa1/0
```

The following output from the **show bfd neighbors** command on DeviceC also now shows only one BFD neighbor for DeviceC in the EIGRP network. The relevant command output is shown in bold in the output.

```
DeviceC# show bfd neighbors

OurAddr      NeighAddr

      LD/RD RH  Holdown(mult)  State      Int
172.16.1.3  172.16.1.1

      3/5  1   114 (3 )      Up        Fa1/0
```

Example: Configuring BFD in an OSPF Network

The following example shows how to configure BFD in an OSPF network. In the following example, a simple OSPF network consists of Device A and Device B. Fast Ethernet interface 0/1 on Device A is connected to the same network as Fast Ethernet interface 6/0 in Device B. The example, starting in global configuration mode, shows the configuration of BFD. For both Devices A and B, BFD is configured globally for all interfaces associated with the OSPF process.

Configuration for Device A

```
!
interface Fast Ethernet 0/1
ip address 172.16.10.1 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
!
interface Fast Ethernet 3/0.1
ip address 172.17.0.1 255.255.255.0
!
router ospf 123
log-adjacency-changes detail
network 172.16.0.0 0.0.0.255 area 0
network 172.17.0.0 0.0.0.255 area 0
bfd all-interfaces
```

Configuration for Device B

```
!
interface Fast Ethernet 6/0
ip address 172.16.10.2 255.255.255.0
```

```

bfd interval 50 min_rx 50 multiplier 3
!
interface Fast Ethernet 6/1
 ip address 172.18.0.1 255.255.255.0
!
router ospf 123
 log-adjacency-changes detail
 network 172.16.0.0 0.0.255.255 area 0
 network 172.18.0.0 0.0.255.255 area 0
 bfd all-interfaces

```

The output from the **show bfd neighbors details** command verifies that a BFD session has been created and that OSPF is registered for BFD support.

Device A

```
DeviceA# show bfd neighbors details
```

```

OurAddr      NeighAddr    LD/RD RH  Holdown(mult)  State    Int
172.16.10.1  172.16.10.2  1/2 1    532 (3 )      Up       Fa0/1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 1000, Received Multiplier: 3
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332 last: 68 ms ago
Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196 last: 192 ms ago
Registered protocols: OSPF

```

```
Uptime: 02:18:49
```

```
Last packet: Version: 0
```

```

- Diagnostic: 0
I Hear You bit: 1      - Demand bit: 0
Poll bit: 0           - Final bit: 0
Multiplier: 3         - Length: 24
My Discr.: 2          - Your Discr.: 1
Min tx interval: 50000 - Min rx interval: 1000
Min Echo interval: 0

```

The output from the **show bfd neighbors details** command from Device B verifies that a BFD session has been created:

Device B

```
DeviceB# attach 6
```

```
Entering Console for 8 Port Fast Ethernet in Slot: 6
```

```
Type "exit" to end this session
```

```
Press RETURN to get started!
```

```
Device> show bfd neighbors details
```

```
Cleanup timer hits: 0
```

```

OurAddr      NeighAddr    LD/RD RH  Holdown(mult)  State    Int
172.16.10.2  172.16.10.1  8/1 1    1000 (5 )      Up       Fa6/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 1000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 5
Holdown (hits): 1000(0), Hello (hits): 200(5995)
Rx Count: 10126, Rx Interval (ms) min/max/avg: 152/248/196 last: 0 ms ago
Tx Count: 5998, Tx Interval (ms) min/max/avg: 204/440/332 last: 12 ms ago
Last packet: Version: 0      - Diagnostic: 0
I Hear You bit: 1          - Demand bit: 0

```



```

Poll bit: 0          - Final bit: 0
Multiplier: 5       - Length: 24
My Discr.: 1        - Your Discr.: 8
Min tx interval: 200000 - Min rx interval: 200000
Min Echo interval: 0
Uptime: 00:33:13
SSO Cleanup Timer called: 0
SSO Cleanup Action Taken: 0
Pseudo pre-emptive process count: 239103 min/max/avg: 8/16/8 last: 0 ms ago
IPC Tx Failure Count: 0
IPC Rx Failure Count: 0
Total Adjs Found: 1

```

The output from the **show ip ospf** command verifies that BFD has been enabled for OSPF.

Device A

```

DeviceA# show ip ospf

Routing Process "ospf 123" with ID 172.16.10.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPF's 10000 msec
Maximum wait time between two consecutive SPF's 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
BFD is enabled

Area BACKBONE(0)
Number of interfaces in this area is 2 (1 loopback)
Area has no authentication
SPF algorithm last executed 00:00:08.828 ago
SPF algorithm executed 9 times
Area ranges are
Number of LSA 3. Checksum Sum 0x028417
Number of opaque link LSA 0. Checksum Sum 0x000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

```

Device B

```

DeviceB# show ip ospf

Routing Process "ospf 123" with ID 172.18.0.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)

```

```

Supports area transit capability
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPF's 10000 msecs
Maximum wait time between two consecutive SPF's 10000 msecs
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Number of external LSA 0. Checksum Sum 0x0
Number of opaque AS LSA 0. Checksum Sum 0x0
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
BFD is enabled

```

```

Area BACKBONE(0)
  Number of interfaces in this area is 2 (1 loopback)
  Area has no authentication
  SPF algorithm last executed 02:07:30.932 ago
  SPF algorithm executed 7 times
  Area ranges are
  Number of LSA 3. Checksum Sum 0x28417
  Number of opaque link LSA 0. Checksum Sum 0x0
  Number of DCbitless LSA 0
  Number of indication LSA 0
  Number of DoNotAge LSA 0
  Flood list length 0

```

The output from the **show ip ospf interface** command verifies that BFD has been enabled for OSPF on the interfaces connecting Device A and Device B.

Device A

```

DeviceA# show ip ospf interface Fast Ethernet 0/1

show ip ospf interface Fast Ethernet 0/1
Fast Ethernet0/1 is up, line protocol is up
Internet Address 172.16.10.1/24, Area 0
Process ID 123, Router ID 172.16.10.1, Network Type BROADCAST, Cost: 1
Transmit Delay is 1 sec, State BDR, Priority 1, BFD enabled
Designated Router (ID) 172.18.0.1, Interface address 172.16.10.2
Backup Designated router (ID) 172.16.10.1, Interface address 172.16.10.1
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
oob-resync timeout 40
Hello due in 00:00:03
Supports Link-local Signaling (LLS)
Index 1/1, flood queue length 0
Next 0x0(0)/0x0(0)
Last flood scan length is 1, maximum is 1
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 1, Adjacent neighbor count is 1
Adjacent with neighbor 172.18.0.1 (Designated Router)
Suppress hello for 0 neighbor(s)

```

Device B

```

DeviceB# show ip ospf interface Fast Ethernet 6/1

```

```

Fast Ethernet6/1 is up, line protocol is up
Internet Address 172.18.0.1/24, Area 0
Process ID 123, Router ID 172.18.0.1, Network Type BROADCAST, Cost: 1
Transmit Delay is 1 sec, State DR, Priority 1, BFD enabled
Designated Router (ID) 172.18.0.1, Interface address 172.18.0.1
No backup designated router on this network
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
oob-resync timeout 40
Hello due in 00:00:01
Supports Link-local Signaling (LLS)
Index 1/1, flood queue length 0
Next 0x0(0)/0x0(0)
Last flood scan length is 0, maximum is 0
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 0, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)

```

Example: Configuring BFD Support for Static Routing

In the following example, the network consists of Device A and Device B. Serial interface 2/0 on Device A is connected to the same network as serial interface 2/0 on Device B. In order for the BFD session to come up, Device B must be configured.

Device A

```

configure terminal
interface Serial 2/0
ip address 10.201.201.1 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.2
ip route 10.0.0.0 255.0.0.0 Serial 2/0 10.201.201.2

```

Device B

```

configure terminal
interface Serial 2/0
ip address 10.201.201.2 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.1
ip route 10.1.1.1 255.255.255.255 Serial 2/0 10.201.201.1

```

Note that the static route on Device B exists solely to enable the BFD session between 10.201.201.1 and 10.201.201.2. If there is no useful static route that needs to be configured, select a prefix that will not affect packet forwarding, for example, the address of a locally configured loopback interface.

In the following example, there is an active static BFD configuration to reach 209.165.200.225 through Ethernet interface 0/0 in the BFD group testgroup. As soon as the static route is configured that is tracked by the configured static BFD, a single hop BFD session is initiated to 209.165.200.225 through Ethernet interface 0/0. The prefix 10.0.0.0/8 is added to the RIB if a BFD session is successfully established.

```

configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225

```

In the following example, a BFD session to 209.165.200.226 through Ethernet interface 0/0.1001 is marked to use the group testgroup. That is, this configuration is a passive static BFD. Though there are static routes

to be tracked by the second static BFD configuration, a BFD session is not triggered for 209.165.200.226 through Ethernet interface 0/0.1001. The existence of the prefixes 10.1.1.1/8 and 10.2.2.2/8 is controlled by the active static BFD session (Ethernet interface 0/0 209.165.200.225).

```
configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
ip route static bfd Ethernet 0/0.1001 209.165.200.226 group testgroup passive
ip route 10.1.1.1 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
ip route 10.2.2.2 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
```



CHAPTER 28

Configuring EtherChannels

- [Finding Feature Information, on page 541](#)
- [Restrictions for EtherChannels, on page 541](#)
- [Information About EtherChannels, on page 542](#)
- [How to Configure EtherChannels, on page 553](#)
- [Monitoring EtherChannel, PAgP, and LACP Status, on page 566](#)
- [Configuration Examples for Configuring EtherChannels, on page 567](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restrictions for EtherChannels

- All ports in an EtherChannel must be assigned to the same VLAN or they must be configured as trunk ports.
- When the ports in an EtherChannel are configured as trunk ports, all the ports must be configured with the same mode (either Inter-Switch Link [ISL] or IEEE 802.1Q).
- Port Aggregation Protocol (PAgP) can be enabled only in single-switch EtherChannel configurations; PAgP cannot be enabled on cross-stack EtherChannels.

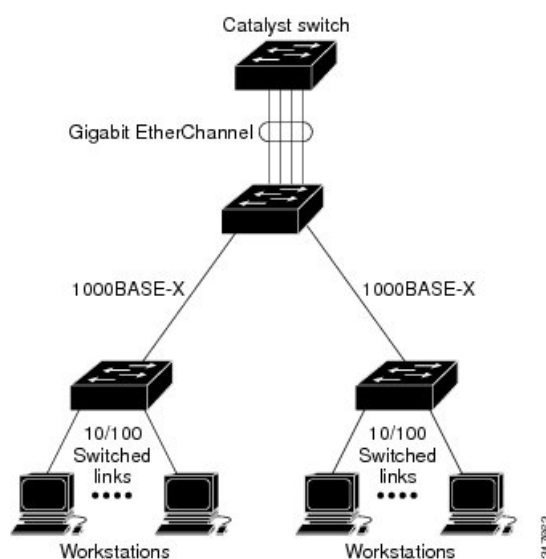
Information About EtherChannels

EtherChannel Overview

EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use the EtherChannel to increase the bandwidth between the wiring closets and the data center, and you can deploy it anywhere in the network where bottlenecks are likely to occur. EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention.

An EtherChannel consists of individual Ethernet links bundled into a single logical link.

Figure 47: Typical EtherChannel Configuration



Each EtherChannel can consist of up to eight compatibly configured Ethernet ports.

EtherChannel Modes

You can configure an EtherChannel in one of these modes: Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), or On. Configure both ends of the EtherChannel in the same mode:

- When you configure one end of an EtherChannel in either PAgP or LACP mode, the system negotiates with the other end of the channel to determine which ports should become active. If the remote port cannot negotiate an EtherChannel, the local port is put into an independent state and continues to carry data traffic as would any other single link. The port configuration does not change, but the port does not participate in the EtherChannel.
- When you configure an EtherChannel in the **on** mode, no negotiations take place. The switch forces all compatible ports to become active in the EtherChannel. The other end of the channel (on the other switch) must also be configured in the **on** mode; otherwise, packet loss can occur.

EtherChannel on Devices

You can create an EtherChannel on a device, on a single device in the stack, or on multiple devices in the stack (known as cross-stack EtherChannel).

EtherChannel Link Failover

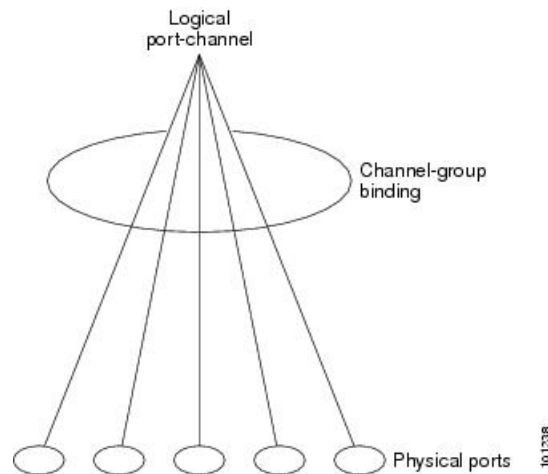
If a link within an EtherChannel fails, traffic previously carried over that failed link moves to the remaining links within the EtherChannel. If traps are enabled on the switch, a trap is sent for a failure that identifies the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

Channel Groups and Port-Channel Interfaces

An EtherChannel comprises a channel group and a port-channel interface. The channel group binds physical ports to the port-channel interface. Configuration changes applied to the port-channel interface apply to all the physical ports bound together in the channel group.

Figure 48: Relationship of Physical Ports, Channel Group and Port-Channel Interface

The **channel-group** command binds the physical port and the port-channel interface together. Each EtherChannel has a port-channel logical interface numbered from 1 to. This port-channel interface number corresponds to the one specified with the **channel-group** interface configuration command.



- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel interface.

You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel interface, but then you must use the **channel-group** *channel-group-number* command to bind the logical interface to a physical port. The *channel-group-number* can be the same as the *port-channel-number*, or you can use a new number. If you use a new number, the **channel-group** command dynamically creates a new port channel.

Port Aggregation Protocol

The Port Aggregation Protocol (PAgP) is a Cisco-proprietary protocol that can be run only on Cisco devices and on those devices licensed by vendors to support PAgP. PAgP facilitates the automatic creation of EtherChannels by exchanging PAgP packets between Ethernet ports.

By using PAgP, the device or device stack learns the identity of partners capable of supporting PAgP and the capabilities of each port. It then dynamically groups similarly configured ports (on a single device in the stack) into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, PAgP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, PAgP adds the group to the spanning tree as a single device port.

PAgP Modes

PAgP modes specify whether a port can send PAgP packets, which start PAgP negotiations, or only respond to PAgP packets received.

Table 51: EtherChannel PAgP Modes

Mode	Description
auto	Places a port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This setting minimizes the transmission of PAgP packets.
desirable	Places a port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets.

Switch ports exchange PAgP packets only with partner ports configured in the **auto** or **desirable** modes. Ports configured in the **on** mode do not exchange PAgP packets.

Both the **auto** and **desirable** modes enable ports to negotiate with partner ports to form an EtherChannel based on criteria such as port speed, and for Layer 2 EtherChannels, based on trunk state and VLAN numbers.

Ports can form an EtherChannel when they are in different PAgP modes as long as the modes are compatible. For example:

- A port in the **desirable** mode can form an EtherChannel with another port that is in the **desirable** or **auto** mode.
- A port in the **auto** mode can form an EtherChannel with another port in the **desirable** mode.

A port in the **auto** mode cannot form an EtherChannel with another port that is also in the **auto** mode because neither port starts PAgP negotiation.

Silent Mode

If your switch is connected to a partner that is PAgP-capable, you can configure the switch port for nonsilent operation by using the **non-silent** keyword. If you do not specify **non-silent** with the **auto** or **desirable** mode, silent mode is assumed.

Use the silent mode when the switch is connected to a device that is not PAgP-capable and seldom, if ever, sends packets. An example of a silent partner is a file server or a packet analyzer that is not generating traffic. In this case, running PAgP on a physical port connected to a silent partner prevents that switch port from ever

becoming operational. However, the silent setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission.

PAgP Learn Method and Priority

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate (logical) ports. The learn method must be configured the same at both ends of the link.

When a device and its partner are both aggregate-port learners, they learn the address on the logical port-channel. The device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.

PAgP cannot automatically detect when the partner device is a physical learner and when the local device is an aggregate-port learner. Therefore, you must manually set the learning method on the local device to learn addresses by physical ports. You also must set the load-distribution method to source-based distribution, so that any given source MAC address is always sent on the same physical port.

You also can configure a single port within the group for all transmissions and use other ports for hot-standby. The unused ports in the group can be swapped into operation in just a few seconds if the selected single port loses hardware-signal detection. You can configure which port is always selected for packet transmission by changing its priority with the **pagp port-priority** interface configuration command. The higher the priority, the more likely that the port will be selected.



Note The device supports address learning only on aggregate ports even though the **physical-port** keyword is provided in the CLI. The **pagp learn-method** command and the **pagp port-priority** command have no effect on the device hardware, but they are required for PAgP interoperability with devices that only support address learning by physical ports, such as the Catalyst 1900 switch.

When the link partner of the device is a physical learner, we recommend that you configure the device as a physical-port learner by using the **pagp learn-method physical-port** interface configuration command. Set the load-distribution method based on the source MAC address by using the **port-channel load-balance src-mac** global configuration command. The device then sends packets to the physical learner using the same port in the EtherChannel from which it learned the source address. Only use the **pagp learn-method** command in this situation.

PAgP Interaction with Virtual Switches and Dual-Active Detection

A virtual switch can be two or more core switches connected by virtual switch links (VSLs) that carry control and data traffic between them. One of the switches is in active mode. The others are in standby mode. For redundancy, remote switches are connected to the virtual switch by remote satellite links (RSLs).

If the VSL between two switches fails, one switch does not know the status of the other. Both switches could change to the active mode, causing a *dual-active situation* in the network with duplicate configurations (including duplicate IP addresses and bridge identifiers). The network might go down.

To prevent a dual-active situation, the core switches send PAgP protocol data units (PDUs) through the RSLs to the remote switches. The PAgP PDUs identify the active switch, and the remote switches forward the PDUs to core switches so that the core switches are in sync. If the active switch fails or resets, the standby switch takes over as the active switch. If the VSL goes down, one core switch knows the status of the other and does not change its state.

PAgP Interaction with Other Features

The Dynamic Trunking Protocol (DTP) and the Cisco Discovery Protocol (CDP) send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive PAgP protocol data units (PDUs) on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

PAgP sends and receives PAgP PDUs only from ports that are up and have PAgP enabled for the auto or desirable mode.

Link Aggregation Control Protocol

The LACP is defined in IEEE 802.3ad and enables Cisco devices to manage Ethernet channels between devices that conform to the IEEE 802.3ad protocol. LACP facilitates the automatic creation of EtherChannels by exchanging LACP packets between Ethernet ports.

By using LACP, the device or device stack learns the identity of partners capable of supporting LACP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, LACP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, LACP adds the group to the spanning tree as a single device port.

The independent mode behavior of ports in a port channel is changed. With CSCtn96950, by default, standalone mode is enabled. When no response is received from an LACP peer, ports in the port channel are moved to suspended state.

LACP Modes

LACP modes specify whether a port can send LACP packets or only receive LACP packets.

Table 52: EtherChannel LACP Modes

Mode	Description
active	Places a port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets.
passive	Places a port into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. This setting minimizes the transmission of LACP packets.

Both the **active** and **passive LACP** modes enable ports to negotiate with partner ports to an EtherChannel based on criteria such as port speed, and for Layer 2 EtherChannels, based on trunk state and VLAN numbers.

Ports can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- A port in the **active** mode can form an EtherChannel with another port that is in the **active** or **passive** mode.

- A port in the **passive** mode cannot form an EtherChannel with another port that is also in the **passive** mode because neither port starts LACP negotiation.

LACP Interaction with Other Features

The DTP and the CDP send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive LACP PDUs on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

LACP sends and receives LACP PDUs only from ports that are up and have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. The **on** mode can be useful if the remote device does not support PAgP or LACP. In the **on** mode, a usable EtherChannel exists only when the devices at both ends of the link are configured in the **on** mode.

Ports that are configured in the **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured in the **on** mode.



Caution

You should use care when using the **on** mode. This is a manual configuration, and ports on both ends of the EtherChannel must have the same configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Load-Balancing and Forwarding Methods

EtherChannel balances the traffic load across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel. You can specify one of several different load-balancing modes, including load distribution based on MAC addresses, IP addresses, source addresses, destination addresses, or both source and destination addresses. The selected mode applies to all EtherChannels configured on the device.



Note

Layer 3 Equal-cost multi path (ECMP) load balancing is based on source IP address, destination IP address, source port, destination port, and layer 4 protocol. Fragmented packets will be treated on two different links based on the algorithm calculated using these parameters. Any changes in one of these parameters will result in load balancing.

MAC Address Forwarding

With source-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the source-MAC address of the incoming packet. Therefore, to provide

load-balancing, packets from different hosts use different ports in the channel, but packets from the same host use the same port in the channel.

With destination-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the destination host's MAC address of the incoming packet. Therefore, packets to the same destination are forwarded over the same port, and packets to a different destination are sent on a different port in the channel.

With source-and-destination MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on both the source and destination MAC addresses. This forwarding method, a combination source-MAC and destination-MAC address forwarding methods of load distribution, can be used if it is not clear whether source-MAC or destination-MAC address forwarding is better suited on a particular device. With source-and-destination MAC-address forwarding, packets sent from host A to host B, host A to host C, and host C to host B could all use different ports in the channel.

IP Address Forwarding

With source-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. To provide load balancing, packets from different IP addresses use different ports in the channel, and packets from the same IP address use the same port in the channel.

With destination-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. To provide load balancing, packets from the same IP source address sent to different IP destination addresses could be sent on different ports in the channel. Packets sent from different source IP addresses to the same destination IP address are always sent on the same port in the channel.

With source-and-destination IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on both the source and destination IP addresses of the incoming packet. This forwarding method, a combination of source-IP and destination-IP address-based forwarding, can be used if it is not clear whether source-IP or destination-IP address-based forwarding is better suited on a particular device. In this method, packets sent from the IP address A to IP address B, from IP address A to IP address C, and from IP address C to IP address B could all use different ports in the channel.

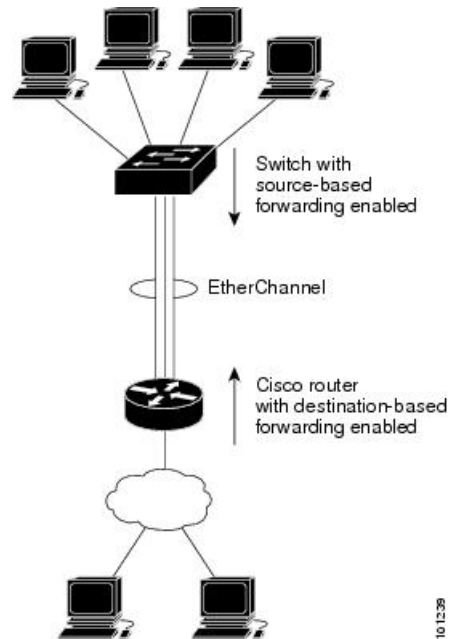
Load-Balancing Advantages

Different load-balancing methods have different advantages, and the choice of a particular load-balancing method should be based on the position of the device in the network and the kind of traffic that needs to be load-distributed.

Figure 49: Load Distribution and Forwarding Methods

In the following figure, an EtherChannel of four workstations communicates with a router. Because the router is a single MAC-address device, source-based forwarding on the device EtherChannel ensures that the device uses all available bandwidth to the router. The router is configured for destination-based forwarding because

the large number of workstations ensures that the traffic is evenly distributed from the router EtherChannel.



Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination-MAC address always chooses the same link in the channel. Using source addresses or IP addresses might result in better load-balancing.

EtherChannel Load Deferral Overview

In an Instant Access system, the EtherChannel Load Deferral feature allows ports to be bundled into port channels, but prevents the assignment of group mask values to these ports. This prevents the traffic from being forwarded to new instant access stack members and reduce data loss following a stateful switchover (SSO).

Cisco Catalyst Instant Access creates a single network touch point and a single point of configuration across distribution and access layer switches. Instant Access enables the merging of physical distribution and access layer switches into a single logical entity with a single point of configuration, management, and troubleshooting. The following illustration represents a sample network where an Instant Access system interacts with a switch (Catalyst 2960-X Series Switches) that is connected via a port channel to stacked clients (Member 1 and Member 2).

When the EtherChannel Load Deferral feature is configured and a new Instant Access client stack member comes up, ports of this newly-joined stack member is bundled into the port channel. In the transition period, the data path is not fully established on the distribution switch (Catalyst 6000 Series Switches), and traffic originating from the access layer switch (Catalyst 2960-X Series Switches) reaches the non-established ports and the traffic gets lost.

When load share deferral is enabled on a port channel, the assignment of a member port's load share is delayed for a period that is configured globally by the **port-channel load-defer** command. During the deferral period, the load share of a deferred member port is set to 0. In this state, the deferred port is capable of receiving data and control traffic, and of sending control traffic, but the port is prevented from sending data traffic to the virtual switching system (VSS). Upon expiration of the global deferral timer, the deferred member port exits the deferral state and the port assumes its normal configured load share.

Load share deferral is applied only if at least one member port of the port channel is currently active with a nonzero load share. If a port enabled for load share deferral is the first member bringing up the EtherChannel, the deferral feature does not apply and the port will forward traffic immediately.

This feature is enabled on a per port-channel basis; however, the load deferral timer is configured globally and not per port-channel. As a result, when a new port is bundled, the timer starts only if it is not already running. If some other ports are already deferred then the new port will be deferred only for the remaining amount of time.

The load deferral is stopped as soon as a member in one of the deferred port channels is unbundled. As a result, all the ports that were deferred is assigned a group-mask in the event of an unbundling during the deferral period.



Note When you try to enable this feature on a stack member switch, the following message is displayed:

```
Load share deferral is supported only on stand-alone stack.
```

Default EtherChannel Configuration

The default EtherChannel configuration is described in this table.

Table 53: Default EtherChannel Configuration

Feature	Default Setting
Channel groups	None assigned.
Port-channel logical interface	None defined.
PAgP mode	No default.
PAgP learn method	Aggregate-port learning on all ports.
PAgP priority	128 on all ports.
LACP mode	No default.
LACP learn method	Aggregate-port learning on all ports.
LACP port priority	32768 on all ports.
LACP system priority	32768.
LACP system ID	LACP system priority and the switch or stack MAC address.
Load-balancing	Load distribution on the switch is based on the source-MAC address of the incoming packet.

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are automatically disabled to avoid network loops and other problems. Follow these guidelines to avoid configuration problems:

- Configure a PAgP EtherChannel with up to eight Ethernet ports of the same type.
- Configure a LACP EtherChannel with up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
- Configure all ports in an EtherChannel to operate at the same speeds and duplex modes.
- Enable all ports in an EtherChannel. A port in an EtherChannel that is disabled by using the **shutdown** interface configuration command is treated as a link failure, and its traffic is transferred to one of the remaining ports in the EtherChannel.
- When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, you must also make the changes to all ports in the group:
 - Allowed-VLAN list
 - Spanning-tree path cost for each VLAN
 - Spanning-tree port priority for each VLAN
 - Spanning-tree Port Fast setting
- Do not configure a port to be a member of more than one EtherChannel group.
- Do not configure an EtherChannel in both the PAgP and LACP modes. EtherChannel groups running PAgP and LACP can coexist on the same device or on different devices in the stack. Individual EtherChannel groups can run either PAgP or LACP, but they cannot interoperate.
- Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x on an EtherChannel port, an error message appears, and IEEE 802.1x is not enabled.
- If EtherChannels are configured on device interfaces, remove the EtherChannel configuration from the interfaces before globally enabling IEEE 802.1x on a device by using the **dot1x system-auth-control** global configuration command.

Layer 2 EtherChannel Configuration Guidelines

When configuring Layer 2 EtherChannels, follow these guidelines:

- Assign all ports in the EtherChannel to the same VLAN, or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.
- An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
- Ports with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.

Auto-LAG

The auto-LAG feature provides the ability to auto create EtherChannels on ports connected to a switch. By default, auto-LAG is disabled globally and is enabled on all port interfaces. The auto-LAG applies to a switch only when it is enabled globally.

On enabling auto-LAG globally, the following scenarios are possible:

- All port interfaces participate in creation of auto EtherChannels provided the partner port interfaces have EtherChannel configured on them. For more information, see the *"The supported auto-LAG configurations between the actor and partner devices"* table below.
- Ports that are already part of manual EtherChannels cannot participate in creation of auto EtherChannels.
- When auto-LAG is disabled on a port interface that is already a part of an auto created EtherChannel, the port interface will unbundle from the auto EtherChannel.

The following table shows the supported auto-LAG configurations between the actor and partner devices:

Table 54: The supported auto-LAG configurations between the actor and partner devices

Actor/Partner	Active	Passive	Auto
Active	Yes	Yes	Yes
Passive	Yes	No	Yes
Auto	Yes	Yes	Yes

On disabling auto-LAG globally, all auto created Etherchannels become manual EtherChannels.

You cannot add any configurations in an existing auto created EtherChannel. To add, you should first convert it into a manual EtherChannel by executing the **port-channel<channel-number>persistent**.



Note Auto-LAG uses the LACP protocol to create auto EtherChannel. Only one EtherChannel can be automatically created with the unique partner devices.

Auto-LAG Configuration Guidelines

Follow these guidelines when configuring the auto-LAG feature.

- When auto-LAG is enabled globally and on the port interface , and if you do not want the port interface to become a member of the auto EtherChannel, disable the auto-LAG on the port interface.
- A port interface will not bundle to an auto EtherChannel when it is already a member of a manual EtherChannel. To allow it to bundle with the auto EtherChannel, first unbundle the manual EtherChannel on the port interface.
- When auto-LAG is enabled and auto EtherChannel is created, you can create multiple EtherChannels manually with the same partner device. But by default, the port tries to create auto EtherChannel with the partner device.
- The auto-LAG is supported only on Layer 2 EtherChannel. It is not supported on Layer 3 interface and Layer 3 EtherChannel.

- The auto-LAG is supported on cross-stack EtherChannel.

How to Configure EtherChannels

After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface, and configuration changes applied to the physical port affect only the port where you apply the configuration.

Configuring Layer 2 EtherChannels

You configure Layer 2 EtherChannels by assigning ports to a channel group with the **channel-group** interface configuration command. This command automatically creates the port-channel logical interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {access | trunk}
4. **switchport access vlan** *vlan-id*
5. **channel-group** *channel-group-number* **mode** {auto [non-silent] | desirable [non-silent] | on } | { active | passive}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/1</pre>	Specifies a physical port, and enters interface configuration mode. Valid interfaces are physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
Step 3	switchport mode {access trunk} Example: <pre>Switch(config-if)# switchport mode access</pre>	Assigns all ports as static-access ports in the same VLAN, or configure them as trunks. If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.

	Command or Action	Purpose
Step 4	<p>switchport access vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Switch(config-if)# switchport access vlan 22</pre>	(Optional) If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.
Step 5	<p>channel-group <i>channel-group-number</i> mode {auto non-silent desirable [non-silent] on } { active passive }</p> <p>Example:</p> <pre>Switch(config-if)# channel-group 5 mode auto</pre>	<p>Assigns the port to a channel group, and specifies the PAgP or the LACP mode.</p> <p>For mode, select one of these keywords:</p> <ul style="list-style-type: none"> • auto —Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation.. • desirable —Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. . • on —Forces the port to channel without PAgP or LACP. In the on mode, an EtherChannel exists only when a port group in the on mode is connected to another port group in the on mode. • non-silent —(Optional) If your device is connected to a partner that is PAgP-capable, configures the device port for nonsilent operation when the port is in the auto or desirable mode. If you do not specify non-silent, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • active—Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • passive —Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation.
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring EtherChannel Load-Balancing

You can configure EtherChannel load-balancing by using source-based or destination-based forwarding methods.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **port-channel load-balance { dst-ip | dst-mac | src-dst-ip | src-dst-mac | src-ip | src-mac }**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	port-channel load-balance { dst-ip dst-mac src-dst-ip src-dst-mac src-ip src-mac } Example: <pre>Switch(config)# port-channel load-balance src-mac</pre>	Configures an EtherChannel load-balancing method. The default is src-mac . Select one of these load-distribution methods: <ul style="list-style-type: none"> • dst-ip—Specifies destination-host IP address. • dst-mac—Specifies the destination-host MAC address of the incoming packet. • src-dst-ip—Specifies the source and destination host IP address. • src-dst-mac—Specifies the source and destination host MAC address. • src-ip—Specifies the source host IP address. • src-mac—Specifies the source MAC address of the incoming packet.
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring Port Channel Load Deferral

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **port-channel load-defer** *seconds*
4. **interface** *type number*
5. **port-channel load-defer**
6. **end**
7. **show etherchannel** *channel-group* **port-channel**
8. **show platform pm group-masks**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	port-channel load-defer <i>seconds</i> Example: Switch(config)# port-channel load-defer 60	Configures the port load share deferral interval for all port channels. <ul style="list-style-type: none">• <i>seconds</i>—The time interval during which load sharing is initially 0 for deferred port channels. The range is 1 to 1800 seconds; the default is 120 seconds
Step 4	interface <i>type number</i> Example: Switch(config)# interface port-channel 10	Configures a port channel interface and enters interface configuration mode.
Step 5	port-channel load-defer Example: Switch(config-if)# port-channel load-defer	Enables port load share deferral on the port channel.
Step 6	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	show etherchannel <i>channel-group</i> port-channel Example: Switch# show etherchannel 1 port-channel	Displays port channel information.

	Command or Action	Purpose
Step 8	show platform pm group-masks Example: Switch# show platform pm group-masks	Display EtherChannel group masks information.

Example

The following is sample output from the **show etherchannel *channel-group* port-channel** command. If the *channel-group* argument is not specified, the command displays information about all channel groups are displayed.

```
Switch# show etherchannel 1 port-channel

Port-channels in the group:
-----

Port-channel: Po1
-----

Age of the Port-channel    = 0d:00h:37m:08s
Logical slot/port        = 9/1           Number of ports = 0
GC                        = 0x00000000   HotStandBy port = null
Port state                = Port-channel Ag-Not-Inuse
Protocol                  = -
Port security             = Disabled
Load share deferral      = Enabled   defer period = 120 sec   time left = 0 sec
```

The following is sample output from the **show platform pm group-masks** command. Deferred ports have the group mask of 0xFFFF, when the defer timer is running.

```
Switch# show platform pm group-masks

=====
Etherchannel members and group masks table
Group #ports group frame-dist slot port mask interface index
-----
1 0 1 src-mac
2 0 2 src-mac
3 0 3 src-mac
4 0 4 src-mac
5 0 5 src-mac
6 0 6 src-mac
7 0 7 src-mac
8 0 8 src-mac
9 0 9 src-mac
10 3 10 src-mac
1 12 0000 Gi1/0/12 3
1 10 FFFF Gi1/0/10 6
1 11 FFFF Gi1/0/11 7
11 0 11 src-mac
12 0 12 src-mac
13 0 13 src-mac
14 0 14 src-mac
15 0 15 src-mac
```

Configuring the PAgP Learn Method and Priority

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **pagp learn-method physical-port**
4. **pagp port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2	Specifies the port for transmission, and enters interface configuration mode.
Step 3	pagp learn-method physical-port Example: Switch(config-if)# pagp learn-method physical port	<p>Selects the PAgP learning method.</p> <p>By default, aggregation-port learning is selected, which means the device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.</p> <p>Selects physical-port to connect with another device that is a physical learner.</p> <p>Make sure to configure the port-channel load-balance global configuration command to src-mac.</p> <p>The learning method must be configured the same at both ends of the link.</p>
Step 4	pagp port-priority <i>priority</i> Example: Switch(config-if)# pagp port-priority 200	<p>Assigns a priority so that the selected port is chosen for packet transmission.</p> <p>For <i>priority</i>, the range is 0 to 255. The default is 128. The higher the priority, the more likely that the port will be used for PAgP transmission.</p>

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring LACP Hot-Standby Ports

When enabled, LACP tries to configure the maximum number of LACP-compatible ports in a channel, up to a maximum of 16 ports. Only eight LACP links can be active at one time. The software places any additional links in a hot-standby mode. If one of the active links becomes inactive, a link that is in the hot-standby mode becomes active in its place.

If you configure more than eight links for an EtherChannel group, the software automatically decides which of the hot-standby ports to make active based on the LACP priority. To every link between systems that operate LACP, the software assigns a unique priority made up of these elements (in priority order):

- LACP system priority
- System ID (the device MAC address)
- LACP port priority
- Port number

In priority comparisons, numerically lower values have higher priority. The priority decides which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

Determining which ports are active and which are hot standby is a two-step procedure. First the system with a numerically lower system priority and system ID is placed in charge of the decision. Next, that system decides which ports are active and which are hot standby, based on its values for port priority and port number. The port priority and port number values for the other system are not used.

You can change the default values of the LACP system priority and the LACP port priority to affect how the software selects active and standby links.

Configuring the LACP System Priority

You can configure the system priority for all the EtherChannels that are enabled for LACP by using the **lacp system-priority** global configuration command. You cannot configure a system priority for each LACP-configured channel. By changing this value from the default, you can affect how the software selects active and standby links.

You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an H port-state flag).

Follow these steps to configure the LACP system priority. This procedure is optional.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **lacp system-priority** *priority*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	lacp system-priority <i>priority</i> Example: Switch(config)# lacp system-priority 32000	Configures the LACP system priority. The range is 1 to 65535. The default is 32768. The lower the value, the higher the system priority.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Configuring the LACP Port Priority

By default, all ports use the same port priority. If the local system has a lower value for the system priority and the system ID than the remote system, you can affect which of the hot-standby links become active first by changing the port priority of LACP EtherChannel ports to a lower value than the default. The hot-standby ports that have lower port numbers become active in the channel first. You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an H port-state flag).



Note If LACP is not able to aggregate all the ports that are compatible (for example, the remote system might have more restrictive hardware limitations), all the ports that cannot be actively included in the EtherChannel are put in the hot-standby state and are used only if one of the channeled ports fails.

Follow these steps to configure the LACP port priority. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lacp port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/2</pre>	Specifies the port to be configured, and enters interface configuration mode.
Step 4	lacp port-priority <i>priority</i> Example: <pre>Switch(config-if)# lacp port-priority 32000</pre>	Configures the LACP port priority. The range is 1 to 65535. The default is 32768. The lower the value, the more likely that the port will be used for LACP transmission.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring the LACP Port Channel Min-Links Feature

You can specify the minimum number of active ports that must be in the link-up state and bundled in an EtherChannel for the port channel interface to transition to the link-up state. Using EtherChannel min-links, you can prevent low-bandwidth LACP EtherChannels from becoming active. Port channel min-links also cause LACP EtherChannels to become inactive if they have too few active member ports to supply the required minimum bandwidth.

To configure the minimum number of links that are required for a port channel. Perform the following tasks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *channel-number*
4. **port-channel min-links** *min-links-number*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface port-channel <i>channel-number</i> Example: Switch(config)# interface port-channel 2	Enters interface configuration mode for a port-channel. For <i>channel-number</i> , the range is 1 to 63.
Step 4	port-channel min-links <i>min-links-number</i> Example: Switch(config-if)# port-channel min-links 3	Specifies the minimum number of member ports that must be in the link-up state and bundled in the EtherChannel for the port channel interface to transition to the link-up state. For <i>min-links-number</i> , the range is 2 to 8.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Configuring LACP Fast Rate Timer

You can change the LACP timer rate to modify the duration of the LACP timeout. Use the **lACP rate** command to set the rate at which LACP control packets are received by an LACP-supported interface. You can change the timeout rate from the default rate (30 seconds) to the fast rate (1 second). This command is supported only on LACP-enabled interfaces.

SUMMARY STEPS

1. enable
2. configure terminal
3. interface { fastethernet | gigabitethernet | tengigabitethernet } slot/port
4. lacp rate { normal | fast }
5. end
6. show lacp internal

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface { fastethernet gigabitethernet tengigabitethernet } slot/port Example: <pre>Switch(config)# interface gigabitEthernet 2/1</pre>	Configures an interface and enters interface configuration mode.
Step 4	lacp rate { normal fast } Example: <pre>Switch(config-if)# lacp rate fast</pre>	Configures the rate at which LACP control packets are received by an LACP-supported interface. <ul style="list-style-type: none"> • To reset the timeout rate to its default, use the no lacp rate command.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show lacp internal Example: <pre>Switch# show lacp internal Switch# show lacp counters</pre>	Verifies your configuration.

Configuring Auto-LAG Globally

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `[no] port-channel auto`
4. `end`
5. `show etherchannel auto`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	[no] port-channel auto Example: Switch(config)# <code>port-channel auto</code>	Enables the auto-LAG feature on a switch globally. Use the no form of this command to disable the auto-LAG feature on the switch globally. Note By default, the auto-LAG feature is enabled on the port.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show etherchannel auto Example: Switch# <code>show etherchannel auto</code>	Displays that EtherChannel is created automatically.

Configuring Auto-LAG on a Port Interface

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`

4. `[no] channel-group auto`
5. `end`
6. `show etherchannel auto`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the port interface to be enabled for auto-LAG, and enters interface configuration mode.
Step 4	[no] channel-group auto Example: <pre>Switch(config-if)# channel-group auto</pre>	(Optional) Enables auto-LAG feature on individual port interface. Use the no form of this command to disable the auto-LAG feature on individual port interface. Note By default, the auto-LAG feature is enabled on the port.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	show etherchannel auto Example: <pre>Switch# show etherchannel auto</pre>	Displays that EtherChannel is created automatically.

What to do next

Configuring Persistence with Auto-LAG

You use the persistence command to convert the auto created EtherChannel into a manual one and allow you to add configuration on the existing EtherChannel.

SUMMARY STEPS

1. `enable`

2. `port-channel channel-number persistent`
3. `show etherchannel summary`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	port-channel channel-number persistent Example: Switch# <code>port-channel 1 persistent</code>	Converts the auto created EtherChannel into a manual one and allows you to add configuration on the EtherChannel.
Step 3	show etherchannel summary Example: Switch# <code>show etherchannel summary</code>	Displays the EtherChannel information.

Monitoring EtherChannel, PAgP, and LACP Status

You can display EtherChannel, PAgP, and LACP status using the commands listed in this table.

Table 55: Commands for Monitoring EtherChannel, PAgP, and LACP Status

Command	Description
<code>clear lacp</code> [<i>channel-group-number</i>] { <code>counters</code> <code>counters</code> }	Clears LACP channel-group information and traffic counters.
<code>clear pagp</code> [<i>channel-group-number</i>] { <code>counters</code> <code>counters</code> }	Clears PAgP channel-group information and traffic counters.
<code>show etherchannel</code> [<i>channel-group-number</i>] { <code>detail</code> <code>load-balance</code> <code>port</code> <code>port-channel</code> <code>protocol</code> <code>summary</code> } [<code>detail</code> <code>load-balance</code> <code>port</code> <code>port-channel</code> <code>protocol</code> <code>auto</code> <code>summary</code>]	Displays EtherChannel information in a brief, detailed, and one-line summary form. Also displays the load-balance or frame-distribution scheme, port, port-channel, protocol, and Auto-LAG information.
<code>show pagp</code> [<i>channel-group-number</i>] { <code>counters</code> <code>internal</code> <code>neighbor</code> }	Displays PAgP information such as traffic information, the internal PAgP configuration, and neighbor information.
<code>show pagp</code> [<i>channel-group-number</i>] <code>dual-active</code>	Displays the dual-active detection status.
<code>show lacp</code> [<i>channel-group-number</i>] { <code>counters</code> <code>internal</code> <code>neighbor</code> <code>sys-id</code> }	Displays LACP information such as traffic information, the internal LACP configuration, and neighbor information.

Command	Description
show running-config	Verifies your configuration entries.
show etherchannel load-balance	Displays the load balance or frame distribution scheme among ports in the port channel.

Configuration Examples for Configuring EtherChannels

Configuring Layer 2 EtherChannels: Examples

This example shows how to configure an EtherChannel on a single device in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode desirable non-silent
Switch(config-if-range)# end
```

This example shows how to configure an EtherChannel on a single device in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

This example shows how to configure a cross-stack EtherChannel. It uses LACP passive mode and assigns two ports on stack member 1 and one port on stack member 2 as static-access ports in VLAN 10 to channel 5:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/4 -5
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode passive
Switch(config-if-range)# exit
Switch(config)# interface gigabitethernet3/0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# channel-group 5 mode passive
Switch(config-if)# exit
```

PoE or LACP negotiation errors may occur if you configure two ports from switch to the access point (AP). This scenario can be avoided if the port channel configuration is on the switch side. For more details, see the following example:

```
interface Port-channel1
  switchport access vlan 20
  switchport mode access
  switchport nonegotiate
  no port-channel standalone-disable  <--this one
  spanning-tree portfast
```



Note If the port reports LACP errors on port flap, you should include the following command as well: **no errdisable detect cause pagp-flap**

Example: Configuring Port Channel Load Deferral

```
Switch# configure terminal
Switch(config)# port-channel load-defer 60
Switch(config)# interface port-channel 10
Switch(config-if)# port-channel load-defer
Switch(config-if)# end
```

Configuring Auto LAG: Examples

This example shows how to configure Auto-LAG on a switch

```
device> enable
device# configure terminal
device (config)# port-channel auto
device (config-if)# end
device# show etherchannel auto
```

The following example shows the summary of EtherChannel that was created automatically.

```
device# show etherchannel auto
Flags: D - down          P - bundled in port-channel
       I - stand-alone  s - suspended
       H - Hot-standby (LACP only)
       R - Layer3       S - Layer2
       U - in use       f - failed to allocate aggregator
       M - not in use, minimum links not met
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port
       A - formed by Auto LAG
```

```
Number of channel-groups in use: 1
Number of aggregators:          1
```

Group	Port-channel	Protocol	Ports
1	Pol(SUA)	LACP	Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)

The following example shows the summary of auto EtherChannel after executing the **port-channel 1 persistent** command.


```

device# port-channel 1 persistent

device# show etherchannel summary
Switch# show etherchannel summary
Flags: D - down          P - bundled in port-channel
       I - stand-alone  s - suspended
       H - Hot-standby (LACP only)
       R - Layer3       S - Layer2
       U - in use       f - failed to allocate aggregator
       M - not in use, minimum links not met
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port
       A - formed by Auto LAG

Number of channel-groups in use: 1
Number of aggregators:          1

Group  Port-channel  Protocol    Ports
-----+-----+-----+-----
1      Po1(SU)        LACP        Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)

```

Configuring LACP Port Channel Min-Links: Examples

This example shows how to configure LACP port-channel min-links:

```

device > enable
device# configure terminal
device(config)# interface port-channel 5
device(config-if)# port-channel min-links 3
device# show etherchannel 25 summary
device# end

```

When the minimum links requirement is not met in standalone switches, the port-channel is flagged and assigned SM/SN or RM/RN state.

```

device# show etherchannel 5 summary

Flags: D - down P - bundled in port-channel
       I - stand-alone s - suspended
       H - Hot-standby (LACP only)
       R - Layer3 S - Layer2
       U - in use N- not in use, no aggregation
       f - failed to allocate aggregator
       M - not in use, no aggregation due to minimum links not met
       m- not in use, port not aggregated due to minimum links not met
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port
Number of channel-groups in use: 6
Number of aggregators: 6

Group  Port-channel  Protocol    Ports
-----+-----+-----+-----
6      Po25(RM)      LACP        Gi1/3/1(D) Gi1/3/2(D) Gi2/2/25(D) Gi2/2/26(W)

```

Example: Configuring LACP Fast Rate Timer

This example shows you how to configure the LACP rate:

```

device> enable
device# configure terminal
device(config)# interface gigabitEthernet 2/1
device(config-if)# lacp rate fast
device(config-if)# exit
device(config)# end
device# show lacp internal
device# show lacp counters

```

The following is sample output from the **show lacp internal** command:

```

device# show lacp internal
Flags: S - Device is requesting Slow LACPDUs
F - Device is requesting Fast LACPDUs
A - Device is in Active mode P - Device is in Passive mode
Channel group 25
LACP port Admin Oper Port Port
Port Flags State Priority Key Key Number State
Te1/49 FA bndl 32768 0x19 0x19 0x32 0x3F
Te1/50 FA bndl 32768 0x19 0x19 0x33 0x3F
Te1/51 FA bndl 32768 0x19 0x19 0x34 0x3F
Te1/52 FA bndl 32768 0x19 0x19 0x35 0x3F

```

The following is sample output from the **show lacp counters** command:

```

device# show lacp counters

LACPDUs Marker Marker Response LACPDUs
Port Sent Recv Sent Recv Sent Recv Pkts Err
-----
Channel group: 24
Te1/1/27 2 2 0 0 0 0 0
Te2/1/25 2 2 0 0 0 0 0

```



CHAPTER 29

Configuring Link-State Tracking

- [Finding Feature Information, on page 571](#)
- [Restrictions for Configuring Link-State Tracking, on page 571](#)
- [Understanding Link-State Tracking, on page 571](#)
- [How to Configure Link-State Tracking, on page 574](#)
- [Monitoring Link-State Tracking, on page 575](#)
- [Configuring Link-State Tracking: Example, on page 575](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restrictions for Configuring Link-State Tracking

- You can configure only two link-state groups per switch.
- An interface cannot be a member of more than one link-state group.
- An interface that is defined as an upstream interface in a link-state group cannot also be defined as a downstream interface in the link-state group.
- Do not enable link-state tracking on individual interfaces that will part of a downstream EtherChannel interface.

Understanding Link-State Tracking

Link-state tracking, also known as trunk failover, binds the link state of multiple interfaces. Link-state tracking can be with server NIC adapter teaming to provide redundancy in the network. When the server NIC adapters

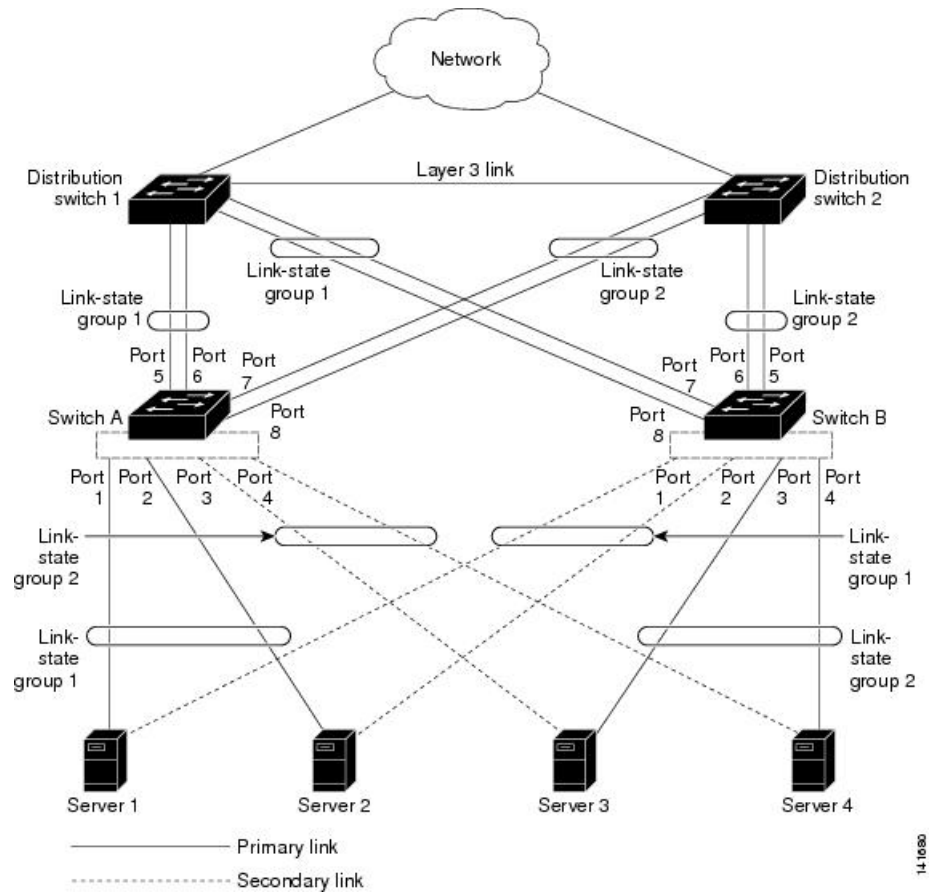
are configured in a primary or secondary relationship, and the link is lost on the primary interface, network connectivity is transparently changed to the secondary interface.



Note An interface can be an aggregation of ports (an EtherChannel) or a single physical port in either access or trunk mode.

The configuration in this figure ensures that the network traffic flow is balanced.

Figure 50: Typical Link-State Tracking Configuration



- For links to switches and other network devices
 - Server 1 and server 2 use switch A for primary links and switch B for secondary links.
 - Server 3 and server 4 use switch B for primary links and switch A for secondary links.
- Link-state group 1 on switch A
 - Switch A provides primary links to server 1 and server 2 through link-state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link-state group 1.

- Port 5 and port 6 are connected to distribution switch 1 through link-state group 1. Port 5 and port 6 are the upstream interfaces in link-state group 1.
- Link-state group 2 on switch A
 - Switch A provides secondary links to server 3 and server 4 through link-state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link-state group 2.
 - Port 7 and port 8 are connected to distribution switch 2 through link-state group 2. Port 7 and port 8 are the upstream interfaces in link-state group 2.
- Link-state group 2 on switch B
 - Switch B provides primary links to server 3 and server 4 through link-state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link-state group 2.
 - Port 5 and port 6 are connected to distribution switch 2 through link-state group 2. Port 5 and port 6 are the upstream interfaces in link-state group 2.
- Link-state group 1 on switch B
 - Switch B provides secondary links to server 1 and server 2 through link-state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link-state group 1.
 - Port 7 and port 8 are connected to distribution switch 1 through link-state group 1. Port 7 and port 8 are the upstream interfaces in link-state group 1.

In a link-state group, the upstream ports can become unavailable or lose connectivity because the distribution switch or router fails, the cables are disconnected, or the link is lost. These are the interactions between the downstream and upstream interfaces when link-state tracking is enabled:

- If any of the upstream interfaces are in the link-up state, the downstream interfaces can change to or remain in the link-up state.
- If all of the upstream interfaces become unavailable, link-state tracking automatically puts the downstream interfaces in the error-disabled state. Connectivity to and from the servers is automatically changed from the primary server interface to the secondary server interface. For example, in the previous figure, if the upstream link for port 6 is lost, the link states of downstream ports 1 and 2 do not change. However, if the link for upstream port 5 is also lost, the link state of the downstream ports changes to the link-down state. Connectivity to server 1 and server 2 is then changed from link-state group 1 to link-state group 2. The downstream ports 3 and 4 do not change state because they are in link-group 2.
- If the link-state group is configured, link-state tracking is disabled, and the upstream interfaces lose connectivity, the link states of the downstream interfaces remain unchanged. The server does not recognize that upstream connectivity has been lost and does not failover to the secondary interface.

You can recover a downstream interface link-down condition by removing the failed downstream port from the link-state group. To recover multiple downstream interfaces, disable the link-state group.

How to Configure Link-State Tracking

To enable link-state tracking, create a link-state group and specify the interfaces that are assigned to the group. This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **link state track** *number*
3. **interface** *interface-id*
4. **link state group** [*number*]{upstream | downstream}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	link state track <i>number</i> Example: Switch(config)# link state track 2	Creates a link-state group and enables link-state tracking. The group number can be 1 or 2; the default is 1.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/1	Specifies a physical interface or range of interfaces to configure, and enters interface configuration mode. Valid interfaces include switch ports in access or trunk mode (IEEE 802.1q) or routed ports. Note Do not enable link-state tracking on individual interfaces that will be part of an Etherchannel interface.
Step 4	link state group [<i>number</i>]{ <u>upstream</u> <u>downstream</u> } Example: Switch(config-if)# link state group 2 upstream	Specifies a link-state group and configures the interface as either an upstream or downstream interface in the group.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Monitoring Link-State Tracking

You can display link-state tracking status using the command in this table.

Table 56: Commands for Monitoring Link-State Tracking Status

Command	Description
<code>show link state group [number] [detail]</code>	Displays the link-state group information.

Configuring Link-State Tracking: Example

This example shows how to create the link-state group 1 and configure the interfaces in the link-state group.

```
Switch# configure terminal
Switch(config)# link state track 1
Switch(config-if)# interface range gigabitethernet1/0/21-22
Switch(config-if)# link state group 1 upstream
Switch(config-if)# interface gigabitethernet1/0/1
Switch(config-if)# link state group 1 downstream
Switch(config-if)# interface gigabitethernet1/0/3
Switch(config-if)# link state group 1 downstream
Switch(config-if)# interface gigabitethernet1/0/5
Switch(config-if)# link state group 1 downstream
Switch(config-if)# end
```




CHAPTER 30

Configuring Resilient Ethernet Protocol

- [Finding Feature Information, on page 577](#)
- [Overview of Resilient Ethernet Protocol, on page 577](#)
- [How to Configure Resilient Ethernet Protocol, on page 582](#)
- [Monitoring Resilient Ethernet Protocol Configuration, on page 591](#)
- [Configuration Examples for Resilient Ethernet Protocol, on page 592](#)
- [Feature Information for Resilient Ethernet Protocol, on page 594](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Overview of Resilient Ethernet Protocol

Resilient Ethernet Protocol (REP) is a Cisco-proprietary protocol that provides an alternative to Spanning Tree Protocol (STP) to control network loops, handle link failures, and improve convergence time. REP controls a group of ports connected in a segment, ensures that the segment does not create any bridging loops, and responds to link failures within the segment. REP provides a basis for constructing more complex networks and supports VLAN load balancing.



Note

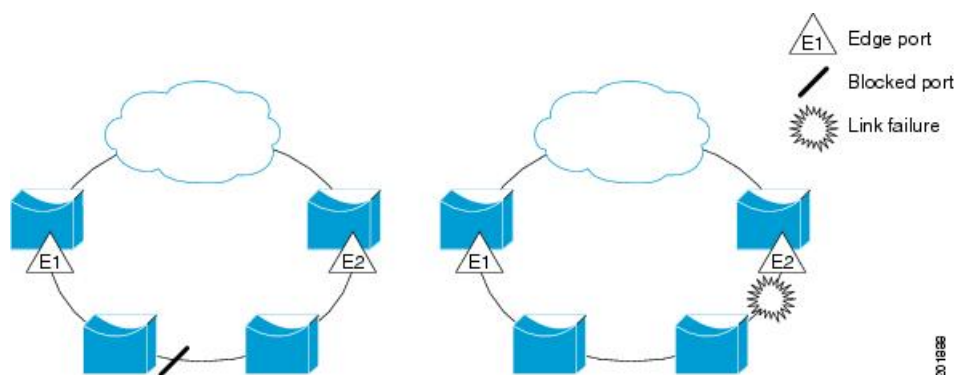
- REP is supported on Catalyst switches running IP Base, IP Lite, and IP Services. REP is not supported on the LAN Base license.
- REP is supported only on Cisco Catalyst 3560-CX switches.

A REP segment is a chain of ports connected to each other and configured with a segment ID. Each segment consists of standard (nonedge) segment ports and two user-configured edge ports. A device can have no more than two ports that belong to the same segment, and each segment port can have only one external neighbor.

A segment can go through a shared medium, but on any link, only two ports can belong to the same segment. REP is supported only on Trunk Ethernet Flow Point (EFP) interfaces.

The following figure shows an example of a segment consisting of six ports spread across four switches. Ports E1 and E2 are configured as edge ports. When all the ports are operational (as in the segment on the left), a single port is blocked, as shown by the diagonal line. When there is a failure in the network, the blocked port returns to the forwarding state to minimize network disruption.

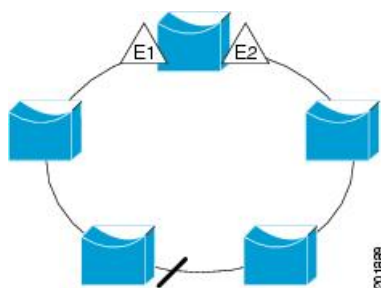
Figure 51: REP Open Segment



The segment shown in the figure above is an open segment; there is no connectivity between the two edge ports. The REP segment cannot cause a bridging loop, and you can safely connect the segment edges to any network. All the hosts connected to devices inside the segment have two possible connections to the rest of the network through the edge ports, but only one connection is accessible at any time. If a failure occurs on any segment or on any port on a REP segment, REP unblocks all the ports to ensure that connectivity is available through the other gateway.

The segment shown in the following figure is a ring segment, with both the edge ports located on the same device. With this configuration, you can create a redundant connection between any two devices in the segment.

Figure 52: REP Ring Segment



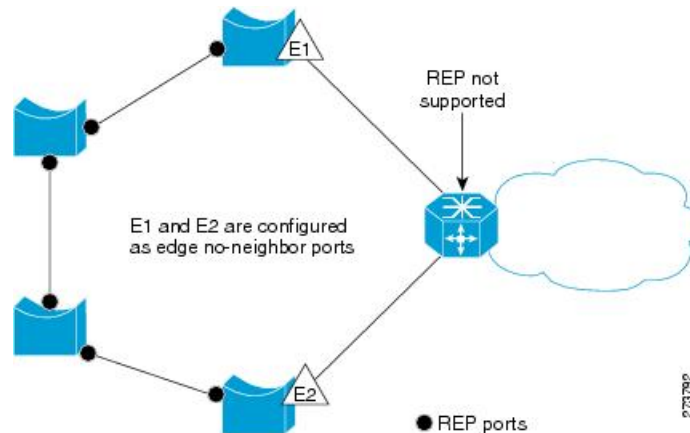
REP segments have the following characteristics:

- If all the ports in a segment are operational, one port (referred to as the *alternate* port) is in the blocked state for each VLAN. If VLAN load balancing is configured, two ports in the segment control the blocked state of VLANs.
- If one or more ports in a segment is not operational, and cause a link failure, all the ports forward traffic on all the VLANs to ensure connectivity.
- In case of a link failure, alternate ports are unblocked as quickly as possible. When the failed link is up, a logically blocked port per VLAN is selected with minimal disruption to the network.

You can construct almost any type of network based on REP segments. REP also supports VLAN load balancing, which is controlled by the primary edge port (any port in the segment).

In access ring-topologies, the neighboring switch might not support REP as shown in the following figure. In this scenario, you can configure the non-REP-facing ports (E1 and E2) as edge no-neighbor ports. These ports inherit all the properties of edge ports, and you can configure them the same as any edge port, including configuring them to send STP or REP topology change notices to the aggregation switch. In this scenario, the STP topology change notice (TCN) that is sent is a multiple spanning-tree (MST) STP message.

Figure 53: Edge No-Neighbor Ports



REP has these limitations:

- You must configure each segment port; an incorrect configuration might cause forwarding loops in the networks.
- REP can manage only a single failed port within the segment; multiple port failures within the REP segment cause loss of network connectivity.
- You should configure REP only in networks with redundancy. Configuring REP in a network without redundancy causes loss of connectivity.

Link Integrity

REP does not use an end-to-end polling function between edge ports to verify link integrity. It implements local link failure detection. The REP Link Status Layer (LSL) detects its REP-aware neighbor and establishes connectivity within the segment. All the VLANs are blocked on an interface until the neighbor is detected. After the neighbor is identified, REP determines which neighbor port should become the alternate port and which ports should forward traffic.

Each port in a segment has a unique port ID. The port ID format is similar to that used by the spanning tree algorithm: a port number (unique on the bridge) associated to a MAC address (unique in the network). When a segment port is coming up, its LSL starts sending packets that include the segment ID and the port ID. The port is declared as operational after it performs a three-way handshake with a neighbor in the same segment.

A segment port does not become operational if:

- No neighbor has the same segment ID.
- More than one neighbor has the same segment ID.

- A neighbor does not acknowledge a local port as a peer.

Each port creates an adjacency with its immediate neighbor. After the neighbor adjacencies are created, the ports negotiate with each other to determine the blocked port for the segment, which will function as the alternate port. All the other ports become unblocked. By default, REP packets are sent to a bridge protocol data unit-class MAC address. The packets can also be sent to a Cisco multicast address, which is used only to send blocked port advertisement (BPA) messages when there is a failure in the segment. The packets are dropped by the devices not running REP.

Fast Convergence

REP runs on a physical link basis and not on a per-VLAN basis. Only one hello message is required for all the VLANs, and this reduces the load on the protocol. We recommend that you create VLANs consistently on all the switches in a given segment and configure the same allowed VLANs on the REP trunk ports. To avoid the delay introduced by relaying messages in software, REP also allows some packets to be flooded to a regular multicast address. These messages operate at the hardware flood layer (HFL) and are flooded to the entire network, not just the REP segment. Switches that do not belong to the segment treat them as data traffic. You can control flooding of these messages by configuring an administrative VLAN for the entire domain or for a particular segment.

The estimated convergence recovery time on fiber interfaces is between 50 ms and 200 ms for the local segment with 200 VLANs configured. Convergence for VLAN load balancing is 300 ms or less.

VLAN Load Balancing

One edge port in the REP segment acts as the primary edge port; and another as the secondary edge port. It is the primary edge port that always participates in VLAN load balancing in the segment. REP VLAN balancing is achieved by blocking some VLANs at a configured alternate port and all the other VLANs at the primary edge port. When you configure VLAN load balancing, you can specify the alternate port in one of three ways:

- By entering the port ID of the interface. To identify the port ID of a port in the segment, enter the **show interface rep detail** interface configuration command for the port.
- By entering the **preferred** keyword to select the port that you previously configured as the preferred alternate port with the **rep segment segment-id preferred** interface configuration command.
- By entering the neighbor offset number of a port in the segment, which identifies the downstream neighbor port of an edge port. The neighbor offset number range is -256 to $+256$; a value of 0 is invalid. The primary edge port has an offset number of 1; positive numbers above 1 identify downstream neighbors of the primary edge port. Negative numbers indicate the secondary edge port (offset number -1) and its downstream neighbors.

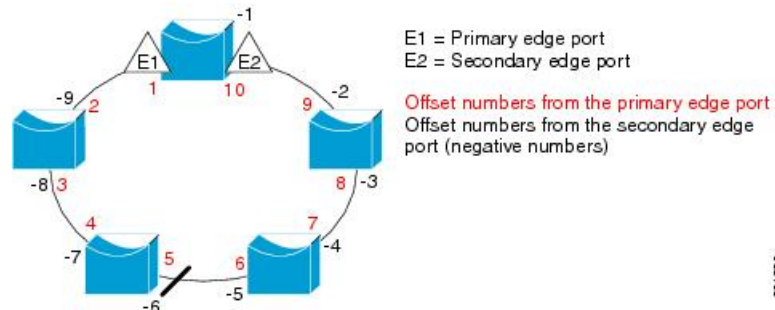


Note Configure offset numbers on the primary edge port by identifying a port's downstream position from the primary (or secondary) edge port. Never enter an offset value of 1 because that is the offset number of the primary edge port.

The following figure shows neighbor offset numbers for a segment, where E1 is the primary edge port and E2 is the secondary edge port. The numbers inside the ring are numbers offset from the primary edge port; the numbers outside of the ring show the offset numbers from the secondary edge port. Note that you can identify all the ports (except the primary edge port) by either a positive offset number (downstream

position from the primary edge port) or a negative offset number (downstream position from the secondary edge port). If E2 became the primary edge port, its offset number would then be 1 and E1 would be -1.

Figure 54: Neighbor Offset Numbers in a Segment



When the REP segment is complete, all the VLANs are blocked. When you configure VLAN load balancing, you must also configure triggers in one of two ways:

- Manually trigger VLAN load balancing at any time by entering the **rep preempt segment** *segment-id* privileged EXEC command on the switch that has the primary edge port.
- Configure a preempt delay time by entering the **rep preempt delay** *seconds* interface configuration command. After a link failure and recovery, VLAN load balancing begins after the configured preemption time period elapses. Note that the delay timer restarts if another port fails before the time has elapsed.



Note When VLAN load balancing is configured, it does not start working until triggered by either manual intervention or a link failure and recovery.

When VLAN load balancing is triggered, the primary edge port sends out a message to alert all the interfaces in the segment about the preemption. When the secondary port receives the message, the message is sent to the network to notify the alternate port to block the set of VLANs specified in the message and to notify the primary edge port to block the remaining VLANs.

You can also configure a particular port in the segment to block all the VLANs. Only the primary edge port initiates VLAN load balancing, which is not possible if the segment is not terminated by an edge port on each end. The primary edge port determines the local VLAN load-balancing configuration.

Reconfigure the primary edge port to reconfigure load balancing. When you change the load-balancing configuration, the primary edge port waits for the **rep preempt segment** command or for the configured preempt delay period after a port failure and recovery, before executing the new configuration. If you change an edge port to a regular segment port, the existing VLAN load-balancing status does not change. Configuring a new edge port might cause a new topology configuration.

Spanning Tree Interaction

REP does not interact with the STP or the Flex Link feature, but can coexist with both. A port that belongs to a segment is removed from spanning tree control, and STP BPDUs are not accepted or sent from segment ports. Therefore, STP cannot run on a segment.

To migrate from an STP ring configuration to an REP segment configuration, begin by configuring a single port in the ring as part of the segment and continue by configuring contiguous ports to minimize the number of segments. Since each segment always contains a blocked port, multiple segments means multiple blocked ports and a potential loss of connectivity. After the segment is configured in both directions up to the location of the edge ports, configure the edge ports.

REP Ports

REP segments consist of Failed, Open, or Alternate ports:

- A port configured as a regular segment port starts as a failed port.
- After the neighbor adjacencies are determined, the port transitions to alternate port state, blocking all the VLANs on the interface. Blocked-port negotiations occur, and when the segment settles, one blocked port remains in the alternate role and all the other ports become open ports.
- When a failure occurs in a link, all the ports move to the Failed state. When the Alternate port receives the failure notification, it changes to the Open state, forwarding all the VLANs.

A regular segment port converted to an edge port, or an edge port converted to a regular segment port, does not always result in a topology change. If you convert an edge port into a regular segment port, VLAN load balancing is not implemented unless it has been configured. For VLAN load balancing, you must configure two edge ports in the segment.

A segment port that is reconfigured as a spanning tree port restarts according to the spanning tree configuration. By default, this is a designated blocking port. If PortFast is configured or if STP is disabled, the port goes into the forwarding state.

How to Configure Resilient Ethernet Protocol

A segment is a collection of ports connected to one another in a chain and configured with a segment ID. To configure REP segments, configure the REP administrative VLAN (or use the default VLAN 1) and then add the ports to the segment, using interface configuration mode. You should configure two edge ports in a segment, with one of them being the primary edge port and the other the secondary edge port by default. A segment should have only one primary edge port. If you configure two ports in a segment as primary edge ports, for example, ports on different switches, the REP selects one of them to serve as the segment's primary edge port. If required, you can configure the location to which segment topology change notices (STCNs) and VLAN load balancing are to be sent.

Default REP Configuration

REP is disabled on all the interfaces. When enabled, the interface is a regular segment port unless it is configured as an edge port.

When REP is enabled, the task of sending segment topology change notices (STCNs) is disabled, all the VLANs are blocked, and the administrative VLAN is VLAN 1.

When VLAN load balancing is enabled, the default is manual preemption with the delay timer disabled. If VLAN load balancing is not configured, the default after manual preemption is to block all the VLANs in the primary edge port.

REP Configuration Guidelines

Follow these guidelines when configuring REP:

- We recommend that you begin by configuring one port and then configure contiguous ports to minimize the number of segments and the number of blocked ports.
- If more than two ports in a segment fail when no external neighbors are configured, one port goes into a forwarding state for the data path to help maintain connectivity during configuration. In the **show rep interface** command output, the Port Role for this port is displayed as **Fail Logical Open**; the Port Role for the other failed port is displayed as **Fail No Ext Neighbor**. When the external neighbors for the failed ports are configured, the ports go through the alternate port transitions and eventually go to an open state, or remain as the alternate port, based on the alternate port selection mechanism.
- REP ports must be Layer 2 IEEE 802.1Q or Trunk ports.
- We recommend that you configure all the trunk ports in a segment with the same set of allowed VLANs.
- Be careful when configuring REP through a Telnet connection because REP blocks all the VLANs until another REP interface sends a message to unblock it. You might lose connectivity to the router if you enable REP in a Telnet session that accesses the router through the same interface.
- You cannot run REP and STP or REP and Flex Links on the same segment or interface.
- If you connect an STP network to an REP segment, be sure that the connection is at the segment edge. An STP connection that is not at the edge might cause a bridging loop because STP does not run on REP segments. All the STP BPDUs are dropped at REP interfaces.
- You must configure all the trunk ports in a segment with the same set of allowed VLANs. If this is not done, misconfiguration occurs.
- If REP is enabled on two ports on a switch, both the ports must be either regular segment ports or edge ports. REP ports follow these rules:
 - There is no limit to the number of REP ports on a switch. However, only two ports on a switch can belong to the same REP segment.
 - If only one port on a switch is configured in a segment, the port should be an edge port.
 - If two ports on a switch belong to the same segment, they must both be edge ports, regular segment ports, or one regular port and one edge no-neighbor port. An edge port and regular segment port on a switch cannot belong to the same segment.
 - If two ports on a switch belong to the same segment, and one is configured as an edge port and one as a regular segment port (a misconfiguration), the edge port is treated as a regular segment port.
- REP interfaces come up in a blocked state and remain in a blocked state until they are safe to be unblocked. You must, therefore, be aware of the status of REP interfaces to avoid sudden connection losses.
- REP sends all the LSL PDUs in the untagged frames to the native VLAN. The BPA message sent to a Cisco multicast address is sent to the administration VLAN, which is VLAN 1 by default.
- You can configure the duration for which a REP interface remains up without receiving a hello from a neighbor. Use the **rep lsl-age-timer** value interface configuration command to set the time from 120 ms to 10000 ms. The LSL hello timer is then set to the age-timer value divided by 3. In normal operation, three LSL hellos are sent before the age timer on the peer switch expires and checks for hello messages.

- EtherChannel port channel interfaces do not support LSL age-timer values less than 1000 ms. If you try to configure a value less than 1000 ms on a port channel, you receive an error message and the command is rejected.
- REP ports cannot be configured as one of the following port types:
 - Switched Port Analyzer (SPAN) destination port
 - Tunnel port
 - Access port
- REP is supported on EtherChannels, but not on an individual port that belongs to an EtherChannel.
- There can be a maximum of 64 REP segments per switch.

Configuring REP Administrative VLAN

To avoid the delay created by link-failure messages, and VLAN-blocking notifications during load balancing, REP floods packets to a regular multicast address at the hardware flood layer (HFL). These messages are flooded to the whole network, and not just the REP segment. You can control the flooding of these messages by configuring an administrative VLAN.

Follow these guidelines when configuring the REP administrative VLAN:

- If you do not configure an administrative VLAN, the default is VLAN 1.
- You can configure one admin VLAN on the switch for all segments.
- The administrative VLAN cannot be the RSPAN VLAN.

To configure the REP administrative VLAN, follow these steps, beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **rep admin vlan** *vlan-id*
3. **end**
4. **show interface** [*interface-id*] **rep detail**
5. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	rep admin vlan <i>vlan-id</i> Example:	Specifies the administrative VLAN. The range is from 2 to 4094.

	Command or Action	Purpose
	<code>Switch(config)# rep admin vlan 2</code>	To set the admin VLAN to 1, which is the default, enter the no rep admin vlan global configuration command.
Step 3	end Example: <code>Switch(config)# end</code>	Exits global configuration mode and returns to privileged EXEC mode.
Step 4	show interface [interface-id] rep detail Example: <code>Switch# show interface gigabitethernet1/1 rep detail</code>	(Optional) Verifies the configuration on a REP interface.
Step 5	copy running-config startup config Example: <code>Switch# copy running-config startup config</code>	(Optional) Saves your entries in the switch startup configuration file.

Configuring a REP Interface

To configure REP, enable REP on each segment interface and identify the segment ID. This task is mandatory, and must be done before other REP configurations. You must also configure a primary and secondary edge port on each segment. All the other steps are optional.

Follow these steps to enable and configure REP on an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **switchport mode trunk**
5. **rep segment segment-id [edge [no-neighbor] [primary]] [preferred]**
6. **rep stcn {interface interface id | segment id-list | stp}**
7. **rep block port {id port-id | neighbor-offset | preferred} vlan {vlan-list | all}**
8. **rep preempt delay seconds**
9. **rep isl-age-timer value**
10. **end**
11. **show interface [interface-id] rep [detail]**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> <code>enable</code>	
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch# <code>interface gigabitethernet1/1</code>	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 4	switchport mode trunk Example: Switch# <code>switchport mode trunk</code>	Configures the interface as a Layer 2 trunk port.
Step 5	rep segment <i>segment-id</i> [edge [no-neighbor] [primary]] [preferred] Example: Switch# <code>rep segment 1 edge no-neighbor primary</code>	<p>Enables REP on the interface and identifies a segment number. The segment ID range is from 1 to 1024.</p> <p>Note You must configure two edge ports, including one primary edge port, for each segment.</p> <p>These optional keywords are available:</p> <ul style="list-style-type: none"> • (Optional) edge—Configures the port as an edge port. Each segment has only two edge ports. Entering the keyword edge without the keyword primary configures the port as the secondary edge port. • (Optional) primary—Configures the port as the primary edge port, the port on which you can configure VLAN load balancing. • (Optional) no-neighbor—Configures a port with no external REP neighbors as an edge port. The port inherits all the properties of an edge port, and you can configure the properties the same way you would for an edge port. <p>Note Although each segment can have only one primary edge port, if you configure edge ports on two different switches and enter the keyword primary on both the switches, the configuration is valid. However, REP selects only one of these ports as the segment primary edge port. You can identify the primary edge port for a segment by entering the show rep topology privileged EXEC command.</p>

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) preferred—Indicates that the port is the preferred alternate port or the preferred port for VLAN load balancing. <p>Note Configuring a port as preferred does not guarantee that it becomes the alternate port; it merely gives the port a slight edge over equal contenders. The alternate port is usually a previously failed port.</p>
Step 6	<p>rep stcn {<i>interface interface id</i> <i>segment id-list</i> stp}</p> <p>Example:</p> <pre>Switch# rep stcn segment 25-50</pre>	<p>(Optional) Configures the edge port to send segment topology change notices (STCNs).</p> <ul style="list-style-type: none"> interface <i>interface-id</i>—Designates a physical interface or port channel to receive STCNs. segment <i>id-list</i>—Identifies one or more segments to receive STCNs. The range is from 1 to 1024. stp—Sends STCNs to STP networks. <p>Note Spanning Tree (MST) mode is required on edge no-neighbor nodes when rep stcn stp command is configured for sending STCNs to STP networks.</p>
Step 7	<p>rep block port {<i>id port-id</i> <i>neighbor-offset</i> preferred} vlan {<i>vlan-list</i> all}</p> <p>Example:</p> <pre>Switch# rep block port id 0009001818D68700 vlan 1-100</pre>	<p>(Optional) Configures VLAN load balancing on the primary edge port, identifies the REP alternate port in one of three ways (id port-id, neighbor_offset, preferred), and configures the VLANs to be blocked on the alternate port.</p> <ul style="list-style-type: none"> id port-id—Identifies the alternate port by port ID. The port ID is automatically generated for each port in the segment. You can view interface port IDs by entering the show interface type number rep [detail] privileged EXEC command. neighbor_offset—Number to identify the alternate port as a downstream neighbor from an edge port. The range is from -256 to 256, with negative numbers indicating the downstream neighbor from the secondary edge port. A value of 0 is invalid. Enter -1 to identify the secondary edge port as the alternate port. <p>Note Because you enter the rep block port command at the primary edge port (offset number 1), you cannot enter an offset value of 1 to identify an alternate port.</p>

	Command or Action	Purpose
		<ul style="list-style-type: none"> • preferred—Selects the regular segment port previously identified as the preferred alternate port for VLAN load balancing. • vlan <i>vlan-list</i>—Blocks one VLAN or a range of VLANs. • vlan all—Blocks all the VLANs. <p>Note Enter this command only on the REP primary edge port.</p>
Step 8	rep preempt delay <i>seconds</i> Example: <pre>Switch# rep preempt delay 100</pre>	(Optional) Configures a preempt time delay. <ul style="list-style-type: none"> • Use this command if you want VLAN load balancing to be automatically triggered after a link failure and recovery. • The time delay range is between 15 to 300 seconds. The default is manual preemption with no time delay. <p>Note Enter this command only on the REP primary edge port.</p>
Step 9	rep lsl-age-timer <i>value</i> Example: <pre>Switch# rep lsl-age-timer 2000</pre>	(Optional) Configures a time (in milliseconds) for which the REP interface remains up without receiving a hello from a neighbor. The range is from 120 to 10000 ms in 40-ms increments. The default is 5000 ms (5 seconds). <p>Note</p> <ul style="list-style-type: none"> • EtherChannel port channel interfaces do not support LSL age-timer values that are less than 1000 ms. • Both the ports on the link should have the same LSL age configured in order to avoid link flaps.
Step 10	end Example: <pre>Switch(config)# end</pre>	Exits global configuration mode and returns to privileged EXEC mode.
Step 11	show interface [<i>interface-id</i>] rep [detail] Example: <pre>Switch(config)# show interface gigabitethernet1/1 rep detail</pre>	(Optional) Displays the REP interface configuration.
Step 12	copy running-config startup-config Example:	(Optional) Saves your entries in the router startup configuration file.

	Command or Action	Purpose
	Switch(config)# copy running-config startup-config	

Setting Manual Preemption for VLAN Load Balancing

If you do not enter the **rep preempt delay seconds** interface configuration command on the primary edge port to configure a preemption time delay, the default is to manually trigger VLAN load balancing on the segment. Be sure that all the other segment configurations have been completed before manually preempting VLAN load balancing. When you enter the **rep preempt delay segment segment-id** command, a confirmation message is displayed before the command is executed because preemption might cause network disruption.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **rep preempt segment segment-id**
4. **show rep topology segment segment-id**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	rep preempt segment segment-id Example: Switch# rep preempt segment 100 The command will cause a momentary traffic disruption. Do you still want to continue? [confirm]	Manually triggers VLAN load balancing on the segment. You need to confirm the command before it is executed.
Step 4	show rep topology segment segment-id Example: Switch# show rep topology segment 100	(Optional) Displays REP topology information.

	Command or Action	Purpose
Step 5	end Example: Switch# end	Exits privileged EXEC mode.

Configuring SNMP Traps for REP

You can configure a router to send REP-specific traps to notify the Simple Network Management Protocol (SNMP) server of link-operational status changes and port role changes.

SUMMARY STEPS

1. **configure terminal**
2. **snmp mib rep trap-rate** *value*
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	snmp mib rep trap-rate <i>value</i> Example: Switch(config)# snmp mib rep trap-rate 500	Enables the switch to send REP traps, and sets the number of traps sent per second. <ul style="list-style-type: none"> • Enter the number of traps sent per second. The range is from 0 to 1000. The default is 0 (no limit is imposed; a trap is sent at every occurrence).
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 4	show running-config Example: Switch# show running-config	(Optional) Displays the running configuration, which can be used to verify the REP trap configuration.
Step 5	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the switch startup configuration file.

Monitoring Resilient Ethernet Protocol Configuration

You can display the rep interface and rep topology details using the commands in this topic.

- **show interface** [*interface-id*] **rep** [**detail**]

Displays REP configuration and status for an interface or for all the interfaces.

- (Optional) **detail**—Displays interface-specific REP information.

Example:

```
Device# show interfaces TenGigabitEthernet4/1 rep detail

TenGigabitEthernet4/1 REP enabled
Segment-id: 3 (Primary Edge)
PortID: 03010015FA66FF80
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 02040015FA66FF804050
Port Role: Open
Blocked VLAN: <empty>
Admin-vlan: 1
Preempt Delay Timer: disabled
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 999, tx: 652
HFL PDU rx: 0, tx: 0
BPA TLV rx: 500, tx: 4
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 6, tx: 5
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 135, tx: 136
```

- **show rep topology** [**segment** *segment-id*] [**archive**] [**detail**]

Displays REP topology information for a segment or for all the segments, including the primary and secondary edge ports in the segment.

- (Optional) **archive**—Displays the last stable topology.



Note An archive topology is not retained when the switch reloads.

- (Optional) **detail**—Displays detailed archived information.

Example:

```
Device# show rep topology

REP Segment 1
BridgeName      PortName      Edge Role
-----
10.64.106.63    Te5/4         Pri  Open
10.64.106.228  Te3/4         Open
10.64.106.228  Te3/3         Open
```

```

10.64.106.67      Te4/3      Open
10.64.106.67      Te4/4      Alt
10.64.106.63      Te4/4      Sec Open

```

```

REP Segment 3
BridgeName      PortName      Edge Role
-----
10.64.106.63    Gi50/1        Pri  Open
SVT_3400_2      Gi0/3          Open
SVT_3400_2      Gi0/4          Open
10.64.106.68    Gi40/2        Open
10.64.106.68    Gi40/1        Open
10.64.106.63    Gi50/2        Sec  Alt

```

Configuration Examples for Resilient Ethernet Protocol

This section provides the following configuration examples:

Example: Configuring the REP Administrative VLAN

This example shows how to configure the administrative VLAN as VLAN 100, and verify the configuration by entering the **show interface rep detail** command on one of the REP interfaces:

```

Switch# configure terminal
Switch(config)# rep admin vlan 100
Switch(config)# end
Switch# show interface gigabitethernet1/1 rep detail

```

```

GigabitEthernet1/1 REP enabled
Segment-id: 2 (Edge)
PortID: 00010019E7144680
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 0002001121A2D5800E4D
Port Role: Open
Blocked Vlan: <empty>
Admin-vlan: 100
Preempt Delay Timer: disabled
LSL Ageout Timer: 5000 ms
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 3322, tx: 1722
HFL PDU rx: 32, tx: 5
BPA TLV rx: 16849, tx: 508
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 118, tx: 118
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 4214, tx: 4190

```

The following example shows how to create an administrative VLAN per segment. Here, VLAN 2 is configured as the administrative VLAN only for REP segment 2. All the remaining segments that are not configured have VLAN 1 as the administrative VLAN by default.

```

Switch# configure terminal
Switch(config)# rep admin vlan 2 segment 2
Switch(config)# end

```


Example: Configuring a REP Interface

This example shows how to configure an interface as the primary edge port for segment 1, to send STCNs to segments 2 through 5, and to configure the alternate port as the port with port ID 0009001818D68700 to block all the VLANs after a preemption delay of 60 seconds after a segment port failure and recovery. The interface is configured to remain up for 6000 ms without receiving a hello from a neighbor.

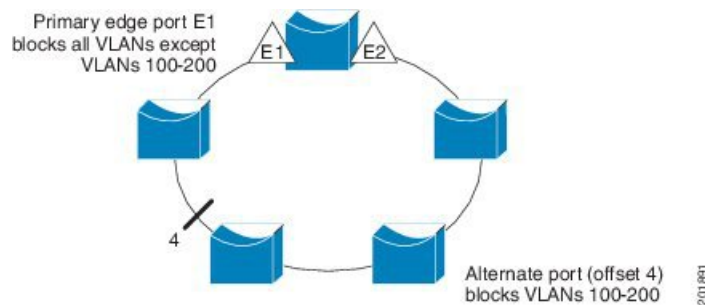
```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
Switch (conf-if)# end
```

This example shows how to configure the same configuration when the interface has no external REP neighbor:

```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge no-neighbor primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
Switch (conf-if)# end
```

This example shows how to configure the VLAN blocking configuration shown in the Figure 5. The alternate port is the neighbor with neighbor offset number 4. After manual preemption, VLANs 100 to 200 are blocked at this port, and all the other VLANs are blocked at the primary edge port E1 (Gigabit Ethernet port 1/1).

Figure 55: Example of VLAN Blocking



```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep block port 4 vlan 100-200
Switch (conf-if)# end
```

Feature Information for Resilient Ethernet Protocol

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 57: Feature Information for Resilient Ethernet Protocol

Feature Name	Release	Feature Information
Resilient Ethernet Protocol	Cisco IOS Release 15.2(6)E1	This feature was introduced. In Cisco IOS Release 15.2(6)E1, this feature is supported on Cisco Catalyst 2960-L Series Switches, Cisco Catalyst 2960-X Series Switches, and Cisco Digital Building.



CHAPTER 31

Configuring Flex Links and the MAC Address-Table Move Update Feature

- [Finding Feature Information, on page 595](#)
- [Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 595](#)
- [Information About Flex Links and MAC Address-Table Move Update, on page 596](#)
- [How to Configure Flex Links and the MAC Address-Table Move Update Feature, on page 599](#)
- [Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 604](#)
- [Configuration Examples for Flex Links, on page 604](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Restrictions for Configuring Flex Links and MAC Address-Table Move Update

- Flex Links are supported only on Layer 2 ports and port channels.
- You can configure up to 16 backup links.
- You can configure only one Flex Links backup link for any active link, and it must be a different interface from the active interface.
- An interface can belong to only one Flex Links pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Links pair.

- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as Flex Links, and you can configure a port channel and a physical interface as Flex Links, with either the port channel or the physical interface as the active link.
- A backup link does not have to be the same type (Gigabit Ethernet or port channel) as the active link. However, you should configure both Flex Links with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on Flex Links ports. A Flex Links port does not participate in STP, even if the VLANs present on the port are configured for STP. When STP is not enabled, be sure that there are no loops in the configured topology.

Information About Flex Links and MAC Address-Table Move Update

Flex Links

Flex Links are a pair of a Layer 2 interfaces (device ports or port channels) where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP). Users can disable STP and still retain basic link redundancy. Flex Links are typically configured in service provider or enterprise networks where customers do not want to run STP on the device. If the device is running STP, Flex Links are not necessary because STP already provides link-level redundancy or backup.

You configure Flex Links on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the Flex Links or backup link. On devices, the Flex Links can be on the same device or on another device in the stack. When one of the links is up and forwarding traffic, the other link is in standby mode, ready to begin forwarding traffic if the other link shuts down. At any given time, only one of the interfaces is in the linkup state and forwarding traffic. If the primary link shuts down, the standby link starts forwarding traffic. When the active link comes back up, it goes into standby mode and does not forward traffic. STP is disabled on Flex Links interfaces.

Flex Links Configuration

In the following figure, ports 1 and 2 on device A are connected to uplink switches B and C. Because they are configured as Flex Links, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

You can also configure a preemption function, specifying the preferred port for forwarding traffic. For example, you can configure the Flex Links pair with preemption mode. In the scenario shown, when port 1 comes back up and has more bandwidth than port 2, port 1 begins forwarding traffic after 60 seconds. Port 2 becomes the standby port. You do this by entering the **switchport backup interface preemption mode bandwidth** and **switchport backup interface preemption delay** interface configuration commands.

If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

Flex Links are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

VLAN Flex Links Load Balancing and Support

VLAN Flex Links load balancing allows users to configure a Flex Links pair so that both ports simultaneously forward the traffic for some mutually exclusive VLANs. For example, if Flex Links ports are configured for 1 to 100 VLANs, the traffic of the first 50 VLANs can be forwarded on one port and the rest on the other port. If one of the ports fail, the other active port forwards all the traffic. When the failed port comes back up, it resumes forwarding traffic in the preferred VLANs. In addition to providing the redundancy, this Flex Links pair can be used for load balancing. Flex Links VLAN load balancing does not impose any restrictions on uplink devices.

Multicast Fast Convergence with Flex Links Failover

Multicast fast convergence reduces the multicast traffic convergence time after a Flex Links failure. Multicast fast convergence is implemented by a combination of learning the backup link as an mrouter port, generating IGMP reports, and leaking IGMP reports.

Learning the Other Flex Links Port as the mrouter Port

In a typical multicast network, there is a querier for each VLAN. A device deployed at the edge of a network has one of its Flex Links ports receiving queries. Flex Links ports are also always forwarding at any given time.

A port that receives queries is added as an mrouter port on the device. An mrouter port is part of all the multicast groups learned by the device. After a changeover, queries are received by the other Flex Links port. The other Flex Links port is then learned as the mrouter port. After changeover, multicast traffic then flows through the other Flex Links port. To achieve faster convergence of traffic, both Flex Links ports are learned as mrouter ports whenever either Flex Links port is learned as the mrouter port. Both Flex Links ports are always part of multicast groups.

Although both Flex Links ports are part of the groups in normal operation mode, all traffic on the backup port is blocked. The normal multicast data flow is not affected by the addition of the backup port as an mrouter port. When the changeover happens, the backup port is unblocked, allowing the traffic to flow. In this case, the upstream multicast data flows as soon as the backup port is unblocked.

Generating IGMP Reports

When the backup link comes up after the changeover, the upstream new distribution device does not start forwarding multicast data, because the port on the upstream router, which is connected to the blocked Flex Links port, is not part of any multicast group. The reports for the multicast groups were not forwarded by the downstream device because the backup link is blocked. The data does not flow on this port, until it learns the multicast groups, which occurs only after it receives reports.

The reports are sent by hosts when a general query is received, and a general query is sent within 60 seconds in normal scenarios. When the backup link starts forwarding, to achieve faster convergence of multicast data, the downstream device immediately sends proxy reports for all the learned groups on this port without waiting for a general query.

Leaking IGMP Reports

To achieve multicast traffic convergence with minimal loss, a redundant data path must be set up before the Flex Links active link goes down. This can be achieved by leaking only IGMP report packets on the Flex Links backup link. These leaked IGMP report messages are processed by upstream distribution routers, so

multicast data traffic gets forwarded to the backup interface. Because all incoming traffic on the backup interface is dropped at the ingress of the access device, no duplicate multicast traffic is received by the host. When the Flex Links active link fails, the access device starts accepting traffic from the backup link immediately. The only disadvantage of this scheme is that it consumes bandwidth on the link between the distribution devices and on the backup link between the distribution and access devices. This feature is disabled by default and can be configured by using the **switchport backup interface *interface-id* multicast fast-convergence** command.

When this feature has been enabled at changeover, the device does not generate the proxy reports on the backup port, which became the forwarding port.

MAC Address-Table Move Update

The MAC address-table move update feature allows the device to provide rapid bidirectional convergence when a primary (forwarding) link goes down and the standby link begins forwarding traffic.

Flex Links VLAN Load Balancing Configuration Guidelines

- For Flex Links VLAN load balancing, you must choose the preferred VLANs on the backup interface.
- You cannot configure a preemption mechanism and VLAN load balancing for the same Flex Links pair.

MAC Address-Table Move Update Configuration Guidelines

- You can enable and configure this feature on the access device to *send* the MAC address-table move updates.
- You can enable and configure this feature on the uplink devices to *get* the MAC address-table move updates.

Default Flex Links and MAC Address-Table Move Update Configuration

- Flex Links is not configured, and there are no backup interfaces defined.
- The preemption mode is off.
- The preemption delay is 35 seconds.
- The MAC address-table move update feature is not configured on the device.

How to Configure Flex Links and the MAC Address-Table Move Update Feature

Configuring Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **switchport backup interface *interface-id***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(conf)# interface gigabitethernet1/0/1	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 3	switchport backup interface <i>interface-id</i> Example: Switch(conf-if)# switchport backup interface gigabitethernet1/0/2	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	end Example: Switch(conf-if)# end	Returns to privileged EXEC mode.

Configuring a Preemption Scheme for a Pair of Flex Links

SUMMARY STEPS

1. **configure terminal**

2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**]
5. **switchport backup interface** *interface-id* **preemption delay** *delay-time*
6. **end**
7. **show interface** [*interface-id*] **switchport backup**
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode
Step 2	interface <i>interface-id</i> Example: <pre>Switch(conf)# interface gigabitethernet1/0/1</pre>	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 3	switchport backup interface <i>interface-id</i> Example: <pre>Switch(conf-if)# switchport backup interface gigabitethernet1/0/2</pre>	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	switchport backup interface <i>interface-id</i> preemption mode [forced bandwidth off] Example: <pre>Switch(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption mode forced</pre>	Configures a preemption mechanism and delay for a Flex Links interface pair. You can configure the preemption as: <ul style="list-style-type: none"> • forced—(Optional) The active interface always preempts the backup. • bandwidth—(Optional) The interface with the higher bandwidth always acts as the active interface. • off—(Optional) No preemption occurs from active to backup.
Step 5	switchport backup interface <i>interface-id</i> preemption delay <i>delay-time</i> Example: <pre>Switch(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption delay 50</pre>	Configures the time delay until a port preempts another port. Note Setting a delay time only works with forced and bandwidth modes.
Step 6	end	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch(conf-if)# end	
Step 7	show interface [<i>interface-id</i>] switchport backup Example: Switch# show interface gigabitethernet1/0/2 switchport backup	Verifies the configuration.
Step 8	copy running-config startup config Example: Switch# copy running-config startup config	(Optional) Saves your entries in the device startup configuration file.

Configuring VLAN Load Balancing on Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id* **prefer vlan** *vlan-range*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch (config)# interface gigabitethernet2/0/6	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 3	switchport backup interface <i>interface-id</i> prefer vlan <i>vlan-range</i> Example: Switch (config-if)# switchport backup interface	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface and specifies the VLANs carried on the interface. The VLAN ID range is 1 to 4094.

	Command or Action	Purpose
	<code>gigabitethernet2/0/8 prefer vlan 2</code>	
Step 4	end Example: Switch (config-if)# end	Returns to privileged EXEC mode.

Configuring MAC Address-Table Move Update

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **switchport backup interface** *interface-id*
 - **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id*
4. **end**
5. **mac address-table move update transmit**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch# interface <code>gigabitethernet1/0/1</code>	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport backup interface <i>interface-id</i> • switchport backup interface <i>interface-id</i> mmu primary vlan <i>vlan-id</i> Example: Switch(config-if)# switchport backup interface	Configures a physical Layer 2 interface (or port channel), as part of a Flex Links pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface. Configure a physical Layer 2 interface (or port channel) and specifies the VLAN ID on the interface, which is used for sending the MAC address-table move update.

	Command or Action	Purpose
	<code>gigabitethernet0/2 mmu primary vlan 2</code>	When one link is forwarding traffic, the other interface is in standby mode.
Step 4	end Example: <code>Switch(config-if) # end</code>	Returns to global configuration mode.
Step 5	mac address-table move update transmit Example: <code>Switch(config) # mac address-table move update transmit</code>	<p>Enables the access device to send MAC address-table move updates to other devices in the network if the primary link goes down and the device starts forwarding traffic through the standby link.</p> <p>Enter command mac address-table move update on the device, for MMU packets to update MAC tables. When the primary link comes back up, the MAC tables need to reconverge and this command will transmit the MMU, that will establish the behavior.</p>
Step 6	end Example: <code>Switch(config) # end</code>	Returns to privileged EXEC mode.

Configuring a Device to Obtain and Process MAC Address-Table Move Update Messages

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table move update receive**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode
Step 2	mac address-table move update receive Example: <code>Switch (config) # mac address-table move update receive</code>	Enables the device to obtain and processes the MAC address-table move updates.

	Command or Action	Purpose
Step 3	end Example: Switch (config)# end	Returns to privileged EXEC mode.

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update

Command	Purpose
show interface [<i>interface-id</i>] switchport backup	Displays the Flex Links backup interface configured for an interface configured Flex Links and the state of each active and backup interface (mode).
show ip igmp profile address-table move update <i>profile-id</i>	Displays the specified IGMP profile or all the IGMP profiles defined.
show mac address-table move update	Displays the MAC address-table move update information on the device.

Configuration Examples for Flex Links

Configuring Flex Links: Examples

This example shows how to verify the configuration after you configure an interface with a backup interface:

```
Switch# show interface switchport backup

Switch Backup Interface Pairs:
Active Interface Backup Interface State
-----
GigabitEthernet1/0/1 GigabitEthernet1/0/2 Active Up/Backup Standby
```

This example shows how to verify the configuration after you configure the preemption mode as forced for a backup interface pair:

```
Switch# show interface switchport backup detail

Switch Backup Interface Pairs:
Active Interface Backup Interface State
-----
GigabitEthernet1/0/211 GigabitEthernet1/0/2 Active Up/Backup Standby
Interface Pair : Gil/0/1, Gil/0/2
Preemption Mode : forced
Preemption Delay : 50 seconds
```

```
Bandwidth : 100000 Kbit (Gi1/0/1), 100000 Kbit (Gi1/0/2)
Mac Address Move Update Vlan : auto
```

Configuring VLAN Load Balancing on Flex Links: Examples

In the following example, VLANs 1 to 50, 60, and 100 to 120 are configured on the device:

```
Switch(config)# interface gigabitethernet 2/0/6
Switch(config-if)# switchport backup interface gigabitethernet 2/0/8 prefer vlan 60,100-120
```

When both interfaces are up, Gi2/0/8 forwards traffic for VLANs 60 and 100 to 120 and Gi2/0/6 forwards traffic for VLANs 1 to 50.

```
Switch# show interfaces switchport backup
```

```
Switch Backup Interface Pairs:
```

Active Interface	Backup Interface	State
GigabitEthernet2/0/6	GigabitEthernet2/0/8	Active Up/Backup Standby

```
Vlans Preferred on Active Interface: 1-50
```

```
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface goes down (LINK_DOWN), VLANs preferred on this interface are moved to the peer interface of the Flex Links pair. In this example, if interface Gi2/0/6 goes down, Gi2/0/8 carries all VLANs of the Flex Links pair.

```
Switch# show interfaces switchport backup
```

```
Switch Backup Interface Pairs:
```

Active Interface	Backup Interface	State
GigabitEthernet2/0/6	GigabitEthernet2/0/8	Active Down/Backup Up

```
Vlans Preferred on Active Interface: 1-50
```

```
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface comes up, VLANs preferred on this interface are blocked on the peer interface and moved to the forwarding state on the interface that has just come up. In this example, if interface Gi2/0/6 comes up, VLANs preferred on this interface are blocked on the peer interface Gi2/0/8 and forwarded on Gi2/0/6.

```
Switch# show interfaces switchport backup
```

```
Switch Backup Interface Pairs:
```

Active Interface	Backup Interface	State
GigabitEthernet2/0/6	GigabitEthernet2/0/8	Active Up/Backup Standby

```
Vlans Preferred on Active Interface: 1-50
```

```
Vlans Preferred on Backup Interface: 60, 100-120
```

```
Switch# show interfaces switchport backup detail

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
FastEthernet1/0/3    FastEthernet1/0/4    Active Down/Backup Up

Vlans Preferred on Active Interface: 1-2,5-4094
Vlans Preferred on Backup Interface: 3-4
Preemption Mode      : off
Bandwidth : 10000 Kbit (Fa1/0/3), 100000 Kbit (Fa1/0/4)
Mac Address Move Update Vlan : auto
```

Configuring the MAC Address-Table Move Update: Examples

This example shows how to verify the configuration after you configure an access device to send MAC address-table move updates:

```
Switch# show mac address-table move update

Switch-ID : 010b.4630.1780
Dst mac-address : 0180.c200.0010
Vlans/Macs supported : 1023/8320
Default/Current settings: Rcv Off/On, Xmt Off/On
Max packets per min : Rcv 40, Xmt 60
Rcv packet count : 5
Rcv conforming packet count : 5
Rcv invalid packet count : 0
Rcv packet count this min : 0
Rcv threshold exceed count : 0
Rcv last sequence# this min : 0
Rcv last interface : Po2
Rcv last src-mac-address : 000b.462d.c502
Rcv last switch-ID : 0403.fd6a.8700
Xmt packet count : 0
Xmt packet count this min : 0
Xmt threshold exceed count : 0
Xmt pak buf unavail cnt : 0
Xmt last interface : None
```

Configuring Multicast Fast Convergence with Flex Links Failover: Examples

These are configuration examples for learning the other Flex Links port as the mrouter port when Flex Links is configured on GigabitEthernet1/0/11 and GigabitEthernet1/0/12, and output for the **show interfaces switchport backup** command:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface GigabitEthernet1/0/11
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport backup interface GigabitEthernet1/0/12
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet1/0/12
Switch(config-if)# switchport trunk encapsulation dot1q
```

```
Switch(config-if)# switchport mode trunk
Switch(config-if)# end
Switch# show interfaces switchport backup detail
Switch Backup Interface Pairs:
Active Interface Backup Interface State
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : Off
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLANs 1 and 401, with their queries reaching the device through GigabitEthernet1/0/11:

```
Switch# show ip igmp snooping querier
```

```
Vlan  IP Address  IGMP Version  Port
-----
1     10.0.0.10   v2           Gi1/0/11
401   41.41.41.1  v2           Gi1/0/11
```

This example is output for the **show ip igmp snooping mrouter** command for VLANs 1 and 401:

```
Switch# show ip igmp snooping mrouter
```

```
Vlan  ports
----  -----
1     Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401   Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

Similarly, both Flex Links ports are part of learned groups. In this example, GigabitEthernet2/0/11 is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Switch# show ip igmp snooping groups
```

```
Vlan  Group    Type    Version  Port List
-----
1     228.1.5.1  igmp   v2       Gi1/0/11, Gi1/0/12, Gi2/0/11
1     228.1.5.2  igmp   v2       Gi1/0/11, Gi1/0/12, Gi2/0/11
```

When a host responds to the general query, the device forwards this report on all the mrouter ports. In this example, when a host sends a report for the group 228.1.5.1, it is forwarded only on GigabitEthernet1/0/11, because the backup port GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, begins forwarding.

As soon as this port starts forwarding, the device sends proxy reports for the groups 228.1.5.1 and 228.1.5.2 on behalf of the host. The upstream router learns the groups and starts forwarding multicast data. This is the default behavior of Flex Links. This behavior changes when the user configures fast convergence using the **switchport backup interface gigabitEthernet 1/0/12 multicast fast-convergence** command. This example shows turning on this feature:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitEthernet 1/0/11
Switch(config-if)# switchport backup interface gigabitEthernet 1/0/12 multicast
fast-convergence
```

```
Switch(config-if)# exit
Switch# show interfaces switchport backup detail

Switch Backup Interface Pairs:
Active      Interface      Backup Interface State
-----
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : On
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLAN 1 and 401 with their queries reaching the device through GigabitEthernet1/0/11:

```
Switch# show ip igmp snooping querier

Vlan  IP Address  IGMP Version  Port
-----
1      10.0.0.10    v2            Gi1/0/11
401    41.41.41.1   v2            Gi1/0/11
```

This is output for the **show ip igmp snooping mrouter** command for VLAN 1 and 401:

```
Switch# show ip igmp snooping mrouter

Vlan  ports
----  -----
1      Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401    Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

Similarly, both the Flex Links ports are a part of the learned groups. In this example, GigabitEthernet2/0/11 is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Switch# show ip igmp snooping groups

Vlan  Group      Type  Version  Port List
-----
1      228.1.5.1  igmp  v2       Gi1/0/11, Gi1/0/12, Gi2/0/11
1      228.1.5.2  igmp  v2       Gi1/0/11, Gi1/0/12, Gi2/0/11
```

Whenever a host responds to the general query, the device forwards this report on all the mrouter ports. When you turn on this feature through the command-line port, and when a report is forwarded by the device on GigabitEthernet1/0/11, it is also leaked to the backup port GigabitEthernet1/0/12. The upstream router learns the groups and starts forwarding multicast data, which is dropped at the ingress because GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, begins forwarding. You do not need to send any proxy reports as the multicast data is already being forwarded by the upstream router. By leaking reports to the backup port, a redundant multicast path has been set up, and the time taken for the multicast traffic convergence is very minimal.



CHAPTER 32

Configuring UniDirectional Link Detection

- [Finding Feature Information, on page 609](#)
- [Restrictions for Configuring UDLD, on page 609](#)
- [Information About UDLD, on page 610](#)
- [How to Configure UDLD, on page 612](#)
- [Monitoring and Maintaining UDLD, on page 614](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

Restrictions for Configuring UDLD

The following are restrictions for configuring UniDirectional Link Detection (UDLD):

- A UDLD-capable port cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another device.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.



Caution

Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Information About UDLD

UniDirectional Link Detection (UDLD) is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it disables the affected port and alerts you. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

Modes of Operation

UDLD has two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected ports on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected ports on fiber-optic links.

In normal and aggressive modes, UDLD works with the Layer 1 mechanisms to learn the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected ports. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

Normal Mode

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic port are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the ports are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In this case, the logical link is considered undetermined, and UDLD does not disable the port.

When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected, as long as autonegotiation is active, the link does not stay up because the Layer 1 mechanisms detects a physical problem with the link. In this case, UDLD does not take any action and the logical link is considered undetermined.

Aggressive Mode

In aggressive mode, UDLD detects a unidirectional link by using the previous detection methods. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of these problems exists:

- On fiber-optic or twisted-pair links, one of the ports cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the ports is down while the other is up.
- One of the fiber strands in the cable is disconnected.

In these cases, UDLD disables the affected port.

In a point-to-point link, UDLD hello packets can be considered as a heart beat whose presence guarantees the health of the link. Conversely, the loss of the heart beat means that the link must be shut down if it is not possible to reestablish a bidirectional link.

If both fiber strands in a cable are working normally from a Layer 1 perspective, UDLD in aggressive mode detects whether those fiber strands are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation because autonegotiation operates at Layer 1.

Methods to Detect Unidirectional Links

UDLD operates by using two methods:

- Neighbor database maintenance
- Event-driven detection and echoing

Neighbor Database Maintenance

UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active port to keep each device informed about its neighbors.

When the device receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the device receives a new hello message before an older cache entry ages, the device replaces the older entry with the new one.

Whenever a port is disabled and UDLD is running, whenever UDLD is disabled on a port, or whenever the device is reset, UDLD clears all existing cache entries for the ports affected by the configuration change. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

Event-Driven Detection and Echoing

UDLD relies on echoing as its detection operation. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the port is disabled.

UDLD Reset Options

If an interface becomes disabled by UDLD, you can use one of the following options to reset UDLD:

- The **udld reset** interface configuration command.
- The **shutdown** interface configuration command followed by the **no shutdown** interface configuration command restarts the disabled port.
- The **no udld {aggressive | enable}** global configuration command followed by the **udld {aggressive | enable}** global configuration command reenables the disabled ports.

- The **no udld port** interface configuration command followed by the **udld port [aggressive]** interface configuration command reenables the disabled fiber-optic port.
- The **errdisable recovery cause udld** global configuration command enables the timer to automatically recover from the UDLD error-disabled state, and the **errdisable recovery interval interval** global configuration command specifies the time to recover from the UDLD error-disabled state.

Default UDLD Configuration

Table 58: Default UDLD Configuration

Feature	Default Setting
UDLD global enable state	Globally disabled
UDLD per-port enable state for fiber-optic media	Disabled on all Ethernet fiber-optic ports
UDLD per-port enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX ports
UDLD aggressive mode	Disabled

How to Configure UDLD

Enabling UDLD Globally

Follow these steps to enable UDLD in the aggressive or normal mode and to set the configurable message timer on all fiber-optic ports on the device.

SUMMARY STEPS

1. **configure terminal**
2. **udld {aggressive | enable | message time *message-timer-interval*}**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	udld {aggressive enable message time <i>message-timer-interval</i>} Example:	Specifies the UDLD mode of operation: <ul style="list-style-type: none"> • aggressive—Enables UDLD in aggressive mode on all fiber-optic ports.

	Command or Action	Purpose
	<pre>Switch(config)# udld enable message time 10</pre>	<ul style="list-style-type: none"> • enable—Enables UDLD in normal mode on all fiber-optic ports on the device. UDLD is disabled by default. An individual interface configuration overrides the setting of the udld enable global configuration command. • message time <i>message-timer-interval</i>—Configures the period of time between UDLD probe messages on ports that are in the advertisement phase and are detected to be bidirectional. The range is from 1 to 90 seconds; the default value is 15. <p>Note This command affects fiber-optic ports only. Use the udld interface configuration command to enable UDLD on other port types.</p> <p>Use the no form of this command, to disable UDLD.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Enabling UDLD on an Interface

Follow these steps either to enable UDLD in the aggressive or normal mode or to disable UDLD on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **udld port [aggressive]**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	interface <i>interface-id</i> Example: Switch(config) # interface gigabitethernet 1/0/1	Specifies the port to be enabled for UDLD, and enters interface configuration mode.
Step 3	udld port [aggressive] Example: Switch(config-if) # udld port aggressive	UDLD is disabled by default. <ul style="list-style-type: none"> • udld port—Enables UDLD in normal mode on the specified port. • udld port aggressive—(Optional) Enables UDLD in aggressive mode on the specified port. <p>Note Use the no udld port interface configuration command to disable UDLD on a specified fiber-optic port.</p>
Step 4	end Example: Switch(config-if) # end	Returns to privileged EXEC mode.

Monitoring and Maintaining UDLD

Command	Purpose
show udld [<i>interface-id</i> neighbors]	Displays the UDLD status for the specified port or for all ports.



PART **V**

Stack Manager and High Availability

- [Configuring HSRP and VRRP, on page 617](#)
- [Configuring Service Level Agreements, on page 637](#)
- [Configuring Enhanced Object Tracking, on page 659](#)
- [Managing Switch Stacks, on page 675](#)



CHAPTER 33

Configuring HSRP and VRRP

- [Configuring HSRP](#) , on page 617

Configuring HSRP

This chapter describes how to use Hot Standby Router Protocol (HSRP) to provide routing redundancy for routing IP traffic without being dependent on the availability of any single router.

You can also use a version of HSRP in Layer 2 mode to configure a redundant command switch to take over cluster management if the cluster command switch fails.



Note HSRP and VRRP features are supported only on Cisco Catalyst 3560-CX switches.

Information About Configuring HSRP

HSRP Overview

HSRP is Cisco's standard method of providing high network availability by providing first-hop redundancy for IP hosts on an IEEE 802 LAN configured with a default gateway IP address. HSRP routes IP traffic without relying on the availability of any single router. It enables a set of router interfaces to work together to present the appearance of a single virtual router or default gateway to the hosts on a LAN. When HSRP is configured on a network or segment, it provides a virtual Media Access Control (MAC) address and an IP address that is shared among a group of configured routers. HSRP allows two or more HSRP-configured routers to use the MAC address and IP network address of a virtual router. The virtual router does not exist; it represents the common target for routers that are configured to provide backup to each other. One of the routers is selected to be the active router and another to be the standby router, which assumes control of the group MAC address and IP address should the designated active router fail.



Note Routers in an HSRP group can be any router interface that supports HSRP, including routed ports and switch virtual interfaces (SVIs).

HSRP provides high network availability by providing redundancy for IP traffic from hosts on networks. In a group of router interfaces, the active router is the router of choice for routing packets; the standby router is the router that takes over the routing duties when an active router fails or when preset conditions are met.

HSRP is useful for hosts that do not support a router discovery protocol and cannot switch to a new router when their selected router reloads or loses power. When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among router interfaces in a group of router interfaces running HSRP. The router selected by the protocol to be the active router receives and routes packets destined for the group's MAC address. For n routers running HSRP, there are $n + 1$ IP and MAC addresses assigned.

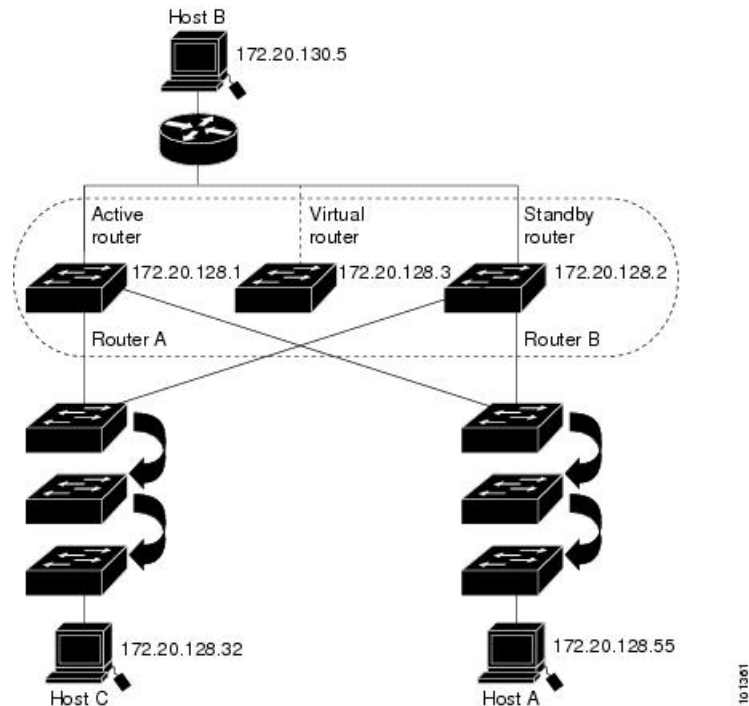
HSRP detects when the designated active router fails, and a selected standby router assumes control of the Hot Standby group's MAC and IP addresses. A new standby router is also selected at that time. Devices running HSRP send and receive multicast UDP-based hello packets to detect router failure and to designate active and standby routers. When HSRP is configured on an interface, Internet Control Message Protocol (ICMP) redirect messages are automatically enabled for the interface.

You can configure multiple Hot Standby groups among switches and switch stacks that are operating in Layer 3 to make more use of the redundant routers.

To do so, specify a group number for each Hot Standby command group you configure for an interface. For example, you might configure an interface on switch 1 as an active router and one on switch 2 as a standby router and also configure another interface on switch 2 as an active router with another interface on switch 1 as its standby router.

The following figure shows a segment of a network configured for HSRP. Each router is configured with the MAC address and IP network address of the virtual router. Instead of configuring hosts on the network with the IP address of Router A, you configure them with the IP address of the virtual router as their default router. When Host C sends packets to Host B, it sends them to the MAC address of the virtual router. If for any reason, Router A stops transferring packets, Router B responds to the virtual IP address and virtual MAC address and becomes the active router, assuming the active router duties. Host C continues to use the IP address of the virtual router to address packets destined for Host B, which Router B now receives and sends to Host B. Until Router A resumes operation, HSRP allows Router B to provide uninterrupted service to users on Host C's segment that need to communicate with users on Host B's segment and also continues to perform its normal function of handling packets between the Host A segment and Host B.

Figure 56: Typical HSRP Configuration



HSRP Versions

and later support these Hot Standby Router Protocol (HSRP) versions:

The switch supports these HSRP versions:

- HSRPv1- Version 1 of the HSRP, the default version of HSRP. It has these features:
 - The HSRP group number can be from 0 to 255.
 - HSRPv1 uses the multicast address 224.0.0.2 to send hello packets, which can conflict with Cisco Group Management Protocol (CGMP) leave processing. You cannot enable HSRPv1 and CGMP at the same time; they are mutually exclusive.
- HSRPv2- Version 2 of the HSRP has these features:
 - HSRPv2 uses the multicast address 224.0.0.102 to send hello packets. HSRPv2 and CGMP leave processing are no longer mutually exclusive, and both can be enabled at the same time.
 - HSRPv2 has a different packet format than HSRPv1.

A switch running HSRPv1 cannot identify the physical router that sent a hello packet because the source MAC address of the router is the virtual MAC address.

HSRPv2 has a different packet format than HSRPv1. A HSRPv2 packet uses the type-length-value (TLV) format and has a 6-byte identifier field with the MAC address of the physical router that sent the packet.

If an interface running HSRPv1 gets an HSRPv2 packet, the type field is ignored.

Multiple HSRP

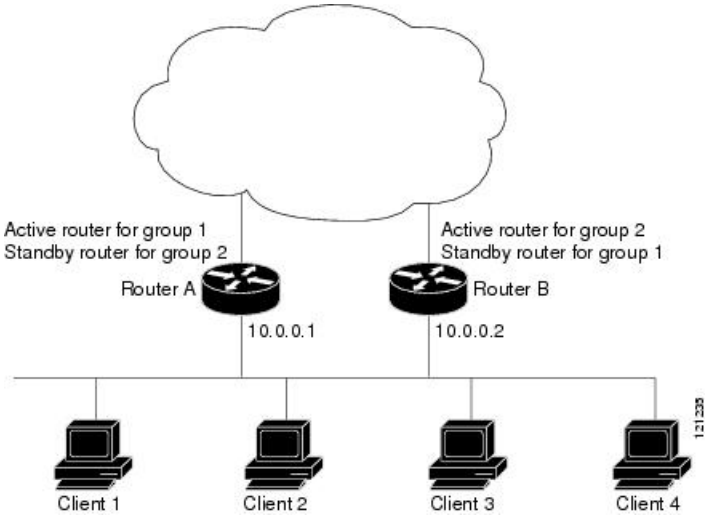
The switch supports Multiple HSRP (MHSRP), an extension of HSRP that allows load sharing between two or more HSRP groups. You can configure MHSRP to achieve load-balancing and to use two or more standby groups (and paths) from a host network to a server network.

In the figure below, half the clients are configured for Router A, and half the clients are configured for Router B. Together, the configuration for Routers A and B establishes two HSRP groups. For group 1, Router A is the default active router because it has the assigned highest priority, and Router B is the standby router. For group 2, Router B is the default active router because it has the assigned highest priority, and Router A is the standby router. During normal operation, the two routers share the IP traffic load. When either router becomes unavailable, the other router becomes active and assumes the packet-transfer functions of the router that is unavailable.



Note For MHSRP, you need to enter the **standby preempt** interface configuration command on the HSRP interfaces so that if a router fails and then comes back up, preemption restores load sharing.

Figure 57: MHSRP Load Sharing



SSO HSRP

SSO HSRP alters the behavior of HSRP when a device with redundant Route Processors (RPs) is configured for stateful switchover (SSO) redundancy mode. When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.

With this functionality, HSRP SSO information is synchronized to the standby RP, allowing traffic that is sent using the HSRP virtual IP address to be continuously forwarded during a switchover without a loss of data or a path change. Additionally, if both RPs fail on the active HSRP device, then the standby HSRP device takes over as the active HSRP device.

The feature is enabled by default when the redundancy mode of operation is set to SSO.

How to Configure HSRP

Default HSRP Configuration

Table 59: Default HSRP Configuration

Feature	Default Setting
HSRP version	Version 1
HSRP groups	None configured
Standby group number	0
Standby MAC address	System assigned as: 0000.0c07.acXX, where XX is the HSRP group number
Standby priority	100
Standby delay	0 (no delay)
Standby track interface priority	10
Standby hello time	3 seconds
Standby holdtime	10 seconds

HSRP Configuration Guidelines

- HSRPv2 and HSRPv1 are mutually exclusive. HSRPv2 is not interoperable with HSRPv1 on an interface and the reverse.
- In the procedures, the specified interface must be one of these Layer 3 interfaces:
 - Routed port: A physical port configured as a Layer 3 port by entering the **no switchport** command in interface configuration mode.
 - SVI: A VLAN interface created by using the **interface vlan** *vlan_id* in global configuration mode, and by default a Layer 3 interface.
 - Etherchannel port channel in Layer 3 mode: A port-channel logical interface created by using the **interface port-channel** *port-channel-number* in global configuration mode, and binding the Ethernet interface into the channel group.
- All Layer 3 interfaces must have IP addresses assigned to them.
- If you change the HSRP version on an interface, each HSRP group resets because it now has a new virtual MAC address.

Enabling HSRP

The **standby ip** interface configuration command activates HSRP on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the address is learned through the standby function. You must configure at least one Layer 3 port on the LAN with the designated address. Configuring an IP address always overrides another designated address currently in use.

When the **standby ip** command is enabled on an interface and proxy ARP is enabled, if the interface's Hot Standby state is active, proxy ARP requests are answered using the Hot Standby group MAC address. If the interface is in a different state, proxy ARP responses are suppressed.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby version** { **1** | **2** }
4. **standby** [*group-number*] **ip** [*ip-address*] [**secondary**]
5. **end**
6. **show standby** [*interface-id*] [*group*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch(config)# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	standby version { 1 2 } Example: Switch(config-if)# standby version 1	(Optional) Configures the HSRP version on the interface. <ul style="list-style-type: none"> • 1- Selects HSRPv1. • 2- Selects HSRPv2. <p>If you do not enter this command or do not specify a keyword, the interface runs the default HSRP version, HSRP v1.</p>
Step 4	standby [<i>group-number</i>] ip [<i>ip-address</i>] [secondary] Example: Switch(config-if)# standby 1 ip	Creates (or enable) the HSRP group using its number and virtual IP address. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is

	Command or Action	Purpose
		designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode
Step 6	show standby [<i>interface-id</i> [<i>group</i>]] Example: Switch # show standby	Verifies the configuration of the standby groups.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring HSRP Priority

The **standby priority**, **standby preempt**, and **standby track** interface configuration commands are all used to set characteristics for finding active and standby routers and behavior regarding when a new active router takes over.

When configuring HSRP priority, follow these guidelines:

- Assigning a priority allows you to select the active and standby routers. If preemption is enabled, the router with the highest priority becomes the active router. If priorities are equal, the current active router does not change.
- The highest number (1 to 255) represents the highest priority (most likely to become the active router).
- When setting the priority, preempt, or both, you must specify at least one keyword (**priority**, **preempt**, or both)
- The priority of the device can change dynamically if an interface is configured with the **standby track** command and another interface on the router goes down.
- The **standby track** interface configuration command ties the router hot standby priority to the availability of its interfaces and is useful for tracking interfaces that are not configured for HSRP. When a tracked interface fails, the hot standby priority on the device on which tracking has been configured decreases by 10. If an interface is not tracked, its state changes do not affect the hot standby priority of the configured device. For each interface configured for hot standby, you can configure a separate list of interfaces to be tracked
- The **standby track interface-priority** interface configuration command specifies how much to decrement the hot standby priority when a tracked interface goes down. When the interface comes back up, the priority is incremented by the same amount.
- When multiple tracked interfaces are down and *interface-priority* values have been configured, the configured priority decrements are cumulative. If tracked interfaces that were not configured with priority values fail, the default decrement is 10, and it is noncumulative.

- When routing is first enabled for the interface, it does not have a complete routing table. If it is configured to preempt, it becomes the active router, even though it is unable to provide adequate routing services. To solve this problem, configure a delay time to allow the router to update its routing table.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP priority characteristics on an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** [*group-number*] **priority***priority*
4. **standby** [*group-number*] **preempt** [**delay** [*minimumseconds*] [*reloadseconds*] [*syncseconds*]]
5. **standby** [*group-number*] **track** *type number* [*interface-priority*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	standby [<i>group-number</i>] priority <i>priority</i> Example: Switch(config-if)# standby 120 priority 50	Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies. Use the no form of the command to restore the default values.
Step 4	standby [<i>group-number</i>] preempt [delay [<i>minimumseconds</i>] [<i>reloadseconds</i>] [<i>syncseconds</i>]] Example: Switch(config-if)# standby 1 preempt delay 300	Configures the router to preempt , which means that when the local router has a higher priority than the active router, it becomes the active router. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies. • (Optional) delay <i>minimum</i>—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 5	standby [<i>group-number</i>] track <i>type number</i> [<i>interface-priority</i>] Example: <pre>Switch(config-if)# standby track interface gigabitethernet1/1/1</pre>	<p>Configures an interface to track other interfaces so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number to which the command applies. • <i>type</i>- Enter the interface type (combined with interface number) that is tracked. • <i>number</i>- Enter the interface number (combined with interface type) that is tracked. • (Optional) <i>interface-priority</i>- Enter the amount by which the hot standby priority for the router is decremented or incremented when the interface goes down or comes back up. The default value is 10.
Step 6	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config	Verifies the configuration of the standby groups.
Step 8	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring MHSRP

To enable MHSRP and load-balancing, you configure two routers as active routers for their groups, with virtual routers as standby routers as shown in the *MHSRP Load Sharing* figure in the Multiple HSRP section. You need to enter the **standby preempt** interface configuration command on each HSRP interface so that if a router fails and comes back up, the preemption occurs and restores load-balancing.

Router A is configured as the active router for group 1, and Router B is configured as the active router for group 2. The HSRP interface for Router A has an IP address of 10.0.0.1 with a group 1 standby priority of 110 (the default is 100). The HSRP interface for Router B has an IP address of 10.0.0.2 with a group 2 standby priority of 110.

Group 1 uses a virtual IP address of 10.0.0.3 and group 2 uses a virtual IP address of 10.0.0.4.

Configuring Router A

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>type number</i> Example: Switch (config)# interface gigabitethernet1/0/1	Configures an interface type and enters interface configuration mode.
Step 3	no switchport Example: Switch (config)# no switchport	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.
Step 4	ip address <i>ip-address mask</i> Example: Switch (config-if)# ip address 10.0.0.1 255.255.255.0	Specifies an IP address for an interface.
Step 5	standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]] Example: Switch (config-if)# standby 1 ip 10.0.0.3	Creates the HSRP group using its number and virtual IP address. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface.

	Command or Action	Purpose
		<p>You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces.</p> <ul style="list-style-type: none"> (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 6	<p>standby [<i>group-number</i>] priority <i>priority</i></p> <p>Example:</p> <pre>Switch(config-if)# standby 1 priority 110</pre>	<p>Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority.</p> <ul style="list-style-type: none"> (Optional) <i>group-number</i>—The group number to which the command applies. <p>Use the no form of the command to restore the default values.</p>
Step 7	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 1 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> (Optional) <i>group-number</i>—The group number to which the command applies. (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 8	<p>standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]]</p> <p>Example:</p> <pre>Switch (config-if)# standby 2 ip 10.0.0.4</pre>	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range

	Command or Action	Purpose
		<p>is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number.</p> <ul style="list-style-type: none"> • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 9	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 2 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>-The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 10	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 11	show running-config	Verifies the configuration of the standby groups.
Step 12	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Router B

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>type number</i> Example: Switch (config)# interface gigabitethernet1/0/1	Configures an interface type and enters interface configuration mode.
Step 3	no switchport Example: Switch (config)# no switchport	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.
Step 4	ip address <i>ip-address mask</i> Example: Switch (config-if)# 10.0.0.2 255.255.255.0	Specifies an IP address for an interface.
Step 5	standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]] Example: Switch (config-if)# standby 1 ip 10.0.0.3	Creates the HSRP group using its number and virtual IP address. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces.

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 6	standby [<i>group-number</i>] priority <i>priority</i> Example: Switch(config-if)# standby 1 priority 110	Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. <ul style="list-style-type: none"> (Optional) <i>group-number</i>—The group number to which the command applies. Use the no form of the command to restore the default values.
Step 7	standby [<i>group-number</i>] preempt [delay [minimum <i>seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]] Example: Switch(config-if)# standby 1 preempt delay 300	Configures the router to preempt , which means that when the local router has a higher priority than the active router, it becomes the active router. <ul style="list-style-type: none"> (Optional) <i>group-number</i>—The group number to which the command applies. (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). Use the no form of the command to restore the default values.
Step 8	standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]] Example: Switch (config-if)# standby 2 ip 10.0.0.4	Creates the HSRP group using its number and virtual IP address. <ul style="list-style-type: none"> (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface.

	Command or Action	Purpose
		<p>You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces.</p> <ul style="list-style-type: none"> (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 9	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 2 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> (Optional) group-number-The group number to which the command applies. (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 10	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 11	show running-config	Verifies the configuration of the standby groups.
Step 12	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring HSRP Authentication and Timers

You can optionally configure an HSRP authentication string or change the hello-time interval and hold-time. When configuring these attributes, follow these guidelines:

- The authentication string is sent unencrypted in all HSRP messages. You must configure the same authentication string on all routers and access servers on a cable to ensure interoperability. Authentication mismatch prevents a device from learning the designated Hot Standby IP address and timer values from other routers configured with HSRP.
- Routers or access servers on which standby timer values are not configured can learn timer values from the active or standby router. The timers configured on an active router always override any other timer settings.
- All routers in a Hot Standby group should use the same timer values. Normally, the *holdtime* is greater than or equal to 3 times the *hellotime*.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP authentication and timers on an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** [*group-number*] **authentication** *string*
4. **standby** [*group-number*] **timers** *hellotime holdtime*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config) # interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	standby [<i>group-number</i>] authentication <i>string</i> Example: Switch(config-if) # standby 1 authentication word	(Optional) authentication <i>string</i> —Enter a string to be carried in all HSRP messages. The authentication string can be up to eight characters in length; the default string is cisco . (Optional) <i>group-number</i> —The group number to which the command applies.
Step 4	standby [<i>group-number</i>] timers <i>hellotime holdtime</i> Example: Switch(config-if) # standby 1 timers 5 15	(Optional) Configure the time interval to send and receive hello packets. • <i>group-number</i> —The group number to which the command applies.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>hellotime</i>—Set the interval between successive hello packets in seconds. The range is 1 to 255 seconds. The default is 3. • <i>holdtime</i>—Set the interval to wait for a hello packet from a neighbor device before declaring the neighbor device as inactive. The range is 1 to 255 seconds. The default is 10.
Step 5	end Example: <pre>Switch(config-if) # end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config	Verifies the configuration of the standby groups.
Step 7	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling HSRP Support for ICMP Redirect Messages

ICMP redirect messages are automatically enabled on interfaces configured with HSRP. ICMP is a network layer Internet protocol that provides message packets to report errors and other information relevant to IP processing. ICMP provides diagnostic functions, such as sending and directing error packets to the host. This feature filters outgoing ICMP redirect messages through HSRP, in which the next hop IP address might be changed to an HSRP virtual IP address. For more information, see the Cisco IOS IP Configuration Guide, Release 12.4.

Configuring HSRP Groups and Clustering

When a device is participating in an HSRP standby routing and clustering is enabled, you can use the same standby group for command switch redundancy and HSRP redundancy. Use the **cluster standby-group HSRP-group-name [routing-redundancy]** global configuration command to enable the same HSRP standby group to be used for command switch and routing redundancy. If you create a cluster with the same HSRP standby group name without entering the **routing-redundancy** keyword, HSRP standby routing is disabled for the group.

Troubleshooting HSRP

If one of the situations as shown in the following table occurs, this message appears:

```
%FHRP group not consistent with already configured groups on the switch stack - virtual MAC reservation failed
```

Table 60: Troubleshooting HSRP

Situation	Action
You configure more than 32 HSRP group instances.	Remove HSRP groups so that up to 32 group instances are configured.

Verifying HSRP

Verifying HSRP Configurations

From privileged EXEC mode, use this command to display HSRP settings:

```
show standby [interface-id [group]] [brief] [detail]
```

You can display HSRP information for the whole switch, for a specific interface, for an HSRP group, or for an HSRP group on an interface. You can also specify whether to display a concise overview of HSRP information or detailed HSRP information. The default display is **detail**. If there are a large number of HSRP groups, using the **show standby** command without qualifiers can result in an unwieldy display.

Example

```
Switch #show standby
VLAN1 - Group 1
Local state is Standby, priority 105, may preempt
Hello time 3 holdtime 10
Next hello sent in 00:00:02.182
Hot standby IP address is 172.20.128.3 configured
Active router is 172.20.128.1 expires in 00:00:09
Standby router is local
Standby virtual mac address is 0000.0c07.ac01
Name is bbb

VLAN1 - Group 100
Local state is Standby, priority 105, may preempt
Hello time 3 holdtime 10
Next hello sent in 00:00:02.262
Hot standby IP address is 172.20.138.51 configured
Active router is 172.20.128.1 expires in 00:00:09
Active router is local
Standby router is unknown expired
Standby virtual mac address is 0000.0c07.ac64
Name is test
```

Configuration Examples for Configuring HSRP

Enabling HSRP: Example

This example shows how to activate HSRP for group 1 on an interface. The IP address used by the hot standby group is learned by using HSRP.



Note This procedure is the minimum number of steps required to enable HSRP. Other configurations are optional.

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 ip
Switch(config-if) # end
Switch # show standby
```

Configuring HSRP Priority: Example

This example activates a port, sets an IP address and a priority of 120 (higher than the default value), and waits for 300 seconds (5 minutes) before attempting to become the active router:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby ip 172.20.128.3
Switch(config-if) # standby priority 120 preempt delay 300
Switch(config-if) # end
Switch # show standby
```

Configuring MHSRP: Example

This example shows how to enable the MHSRP configuration shown in the figure *MHSRP Load Sharing*

Router A Configuration

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # ip address 10.0.0.1 255.255.255.0
Switch(config-if) # standby ip 10.0.0.3
Switch(config-if) # standby 1 priority 110
Switch(config-if) # standby 1 preempt
Switch(config-if) # standby 2 ip 10.0.0.4
Switch(config-if) # standby 2 preempt
Switch(config-if) # end
```

Router B Configuration

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # ip address 10.0.0.2 255.255.255.0
Switch(config-if) # standby ip 10.0.0.3
Switch(config-if) # standby 1 preempt
Switch(config-if) # standby 2 ip 10.0.0.4
Switch(config-if) # standby 1 priority 110
Switch(config-if) # standby 2 preempt
Switch(config-if) # end
```

Configuring HSRP Authentication and Timer: Example

This example shows how to configure word as the authentication string required to allow Hot Standby routers in group 1 to interoperate:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 authentication word
Switch(config-if) # end
```

This example shows how to set the timers on standby group 1 with the time between hello packets at 5 seconds and the time after which a router is considered down to be 15 seconds:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 ip
Switch(config-if) # standby 1 timers 5 15
Switch(config-if) # end
```

Configuring HSRP Groups and Clustering: Example

This example shows how to bind standby group my_hsrp to the cluster and enable the same HSRP group to be used for command switch redundancy and router redundancy. The command can only be executed on the cluster command switch. If the standby group name or number does not exist, or if the switch is a cluster member switch, an error message appears.

```
Switch # configure terminal
Switch(config) # cluster standby-group my_hsrp routing-redundancy
Switch(config-if) # end
```

Information About VRRP

Configuring VRRP

Virtual Router Redundancy Protocol (VRRP) is an election protocol that enables a group of routers to form a single virtual router to provide redundancy. In a VRRP configuration, one router is elected as the virtual router primary, and the other routers act as backups in case it fails. The LAN clients can then be configured with the virtual router as their default gateway, allowing several routers on a multi-access link to use the same virtual IP address. The virtual router, representing a group of routers, forms a VRRP group.

Both HSRP and VRRP perform the same function. You can choose to configure either IETF standard VRRP or Cisco's more powerful HSRP protocol on a device or stack.

Restrictions for VRRP

- The VRRP implementation on the switch does not support the MIB specified in RFC 2787.
- The VRRP implementation on the switch supports only text-based authentication.



CHAPTER 34

Configuring Service Level Agreements

This chapter describes how to use Cisco IOS IP Service Level Agreements (SLAs) on the switch.

Unless otherwise noted, the term *switch* refers to a standalone switch or a switch stack.

- [Finding Feature Information, on page 637](#)
- [Restrictions on SLAs, on page 637](#)
- [Information About SLAs, on page 638](#)
- [How to Configure IP SLAs Operations, on page 642](#)
- [Monitoring IP SLA Operations, on page 655](#)
- [Monitoring IP SLA Operation Examples, on page 656](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restrictions on SLAs

This section lists the restrictions on SLAs.

The following are restrictions on IP SLAs network performance measurement:

- The device does not support VoIP service levels using the gatekeeper registration delay operations measurements.
- Only a Cisco IOS device can be a source for a destination IP SLAs responder.
- You cannot configure the IP SLAs responder on non-Cisco devices and Cisco IOS IP SLAs can send operational packets only to services native to those devices.

Information About SLAs

Cisco IOS IP Service Level Agreements (SLAs)

Cisco IOS IP SLAs send data across the network to measure performance between multiple network locations or across multiple network paths. They simulate network data and IP services and collect network performance information in real time. Cisco IOS IP SLAs generate and analyze traffic either between Cisco IOS devices or from a Cisco IOS device to a remote IP device such as a network application server. Measurements provided by the various Cisco IOS IP SLA operations can be used for troubleshooting, for problem analysis, and for designing network topologies.

Depending on the specific Cisco IOS IP SLA operations, various network performance statistics are monitored within the Cisco device and stored in both command-line interface (CLI) and Simple Network Management Protocol (SNMP) MIBs. IP SLA packets have configurable IP and application layer options such as source and destination IP address, User Datagram Protocol (UDP)/TCP port numbers, a type of service (ToS) byte (including Differentiated Services Code Point [DSCP] and IP Precedence bits), Virtual Private Network (VPN) routing/forwarding instance (VRF), and URL web address.

Because Cisco IP SLAs are Layer 2 transport independent, you can configure end-to-end operations over disparate networks to best reflect the metrics that an end user is likely to experience. IP SLAs collect and analyze the following performance metrics:

- Delay (both round-trip and one-way)
- Jitter (directional)
- Packet loss (directional)
- Packet sequencing (packet ordering)
- Path (per hop)
- Connectivity (directional)
- Server or website download time

Because Cisco IOS IP SLAs is SNMP-accessible, it can also be used by performance-monitoring applications like Cisco Prime Internetwork Performance Monitor (IPM) and other third-party Cisco partner performance management products.

Using IP SLAs can provide the following benefits:

- Service-level agreement monitoring, measurement, and verification.
- Network performance monitoring
 - Measurement of jitter, latency, or packet loss in the network.
 - Continuous, reliable, and predictable measurements.
- IP service network health assessment to verify that the existing QoS is sufficient for new IP services.
- Edge-to-edge network availability monitoring for proactive verification and connectivity testing of network resources (for example, shows the network availability of an NFS server used to store business critical data from a remote site).

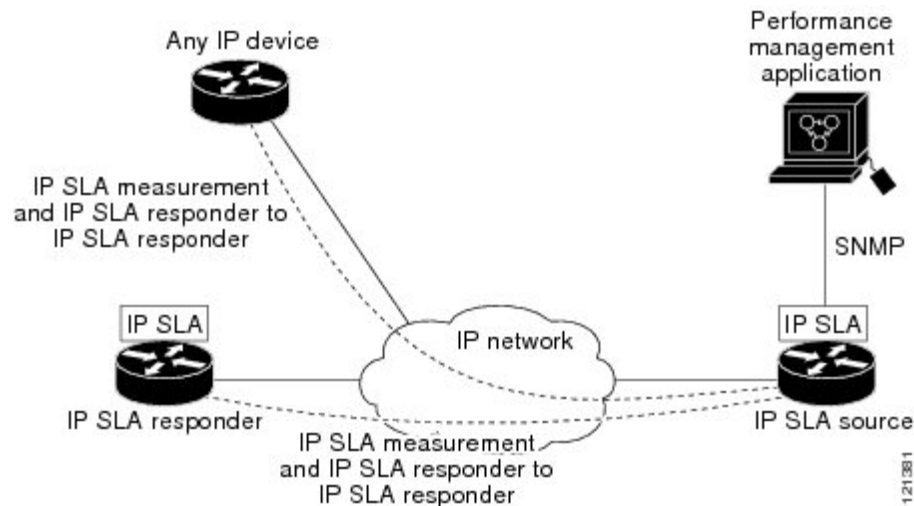
- Network operation troubleshooting by providing consistent, reliable measurement that immediately identifies problems and saves troubleshooting time.
- Multiprotocol Label Switching (MPLS) performance monitoring and network verification (if the device supports MPLS).

Network Performance Measurement with Cisco IOS IP SLAs

You can use IP SLAs to monitor the performance between any area in the network—core, distribution, and edge—without deploying a physical probe. It uses generated traffic to measure network performance between two networking devices.

Figure 58: Cisco IOS IP SLAs Operation

The following figure shows how IP SLAs begin when the source device sends a generated packet to the destination device. After the destination device receives the packet, depending on the type of IP SLAs operation, it responds with time-stamp information for the source to make the calculation on performance metrics. An IP SLAs operation performs a network measurement from the source device to a destination in the network using a specific protocol such as UDP.



IP SLA Responder and IP SLA Control Protocol

The IP SLA responder is a component embedded in the destination Cisco device that allows the system to anticipate and respond to IP SLA request packets. The responder provides accurate measurements without the need for dedicated probes. The responder uses the Cisco IOS IP SLA Control Protocol to provide a mechanism through which it can be notified on which port it should listen and respond.

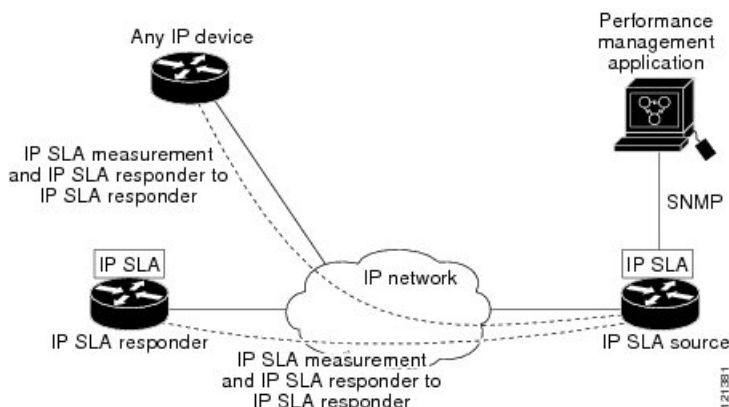


Note The IP SLA responder can be a Cisco IOS Layer 2, responder-configurable device. The responder does not need to support full IP SLA functionality.

The following figure shows where the Cisco IOS IP SLA responder fits in the IP network. The responder listens on a specific port for control protocol messages sent by an IP SLA operation. Upon receipt of the control message, it enables the specified UDP or TCP port for the specified duration. During this time, the

responder accepts the requests and responds to them. It disables the port after it responds to the IP SLA packet, or when the specified time expires. MD5 authentication for control messages is available for added security.

Figure 59: Cisco IOS IP SLAs Operation



You do not need to enable the responder on the destination device for all IP SLA operations. For example, a responder is not required for services that are already provided by the destination router (such as Telnet or HTTP).

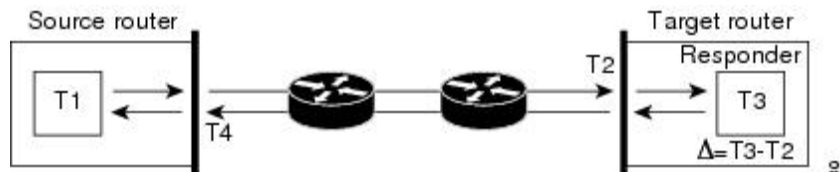
Response Time Computation for IP SLAs

Switches, controllers, and routers can take tens of milliseconds to process incoming packets due to other high priority processes. This delay affects the response times because the test-packet reply might be in a queue while waiting to be processed. In this situation, the response times would not accurately represent true network delays. IP SLAs minimize these processing delays on the source device as well as on the target device (if the responder is being used) to determine true round-trip times. IP SLA test packets use time stamping to minimize the processing delays.

When the IP SLA responder is enabled, it allows the target device to take time stamps when the packet arrives on the interface at interrupt level and again just as it is leaving, eliminating the processing time. This time stamping is made with a granularity of sub-milliseconds (ms).

Figure 60: Cisco IOS IP SLA Responder Time Stamping

The following figure demonstrates how the responder works. Four time stamps are taken to make the calculation for round-trip time. At the target router, with the responder functionality enabled, time stamp 2 (TS2) is subtracted from time stamp 3 (TS3) to produce the time spent processing the test packet as represented by delta. This delta value is then subtracted from the overall round-trip time. Notice that the same principle is applied by IP SLAs on the source router where the incoming time stamp 4 (TS4) is also taken at the interrupt



level to allow for greater accuracy. RTT (Round-trip time) = $T4$ (Time stamp 4) - $T1$ (Time stamp 1) - Δ

An additional benefit of the two time stamps at the target device is the ability to track one-way delay, jitter, and directional packet loss. Because much network behavior is asynchronous, it is critical to have these statistics. However, to capture one-way delay measurements, you must configure both the source router and

target router with Network Time Protocol (NTP) so that the source and target are synchronized to the same clock source. One-way jitter measurements do not require clock synchronization.

IP SLAs Operation Scheduling

When you configure an IP SLAs operation, you must schedule the operation to begin capturing statistics and collecting error information. You can schedule an operation to start immediately or to start at a certain month, day, and hour. You can use the *pending* option to set the operation to start at a later time. The pending option is an internal state of the operation that is visible through SNMP. The pending state is also used when an operation is a reaction (threshold) operation waiting to be triggered. You can schedule a single IP SLAs operation or a group of operations at one time.

You can schedule several IP SLAs operations by using a single command through the Cisco IOS CLI or the CISCO RTTMON-MIB. Scheduling the operations to run at evenly distributed times allows you to control the amount of IP SLAs monitoring traffic. This distribution of IP SLA operations helps minimize the CPU utilization and thus improves network scalability.

For more details about the IP SLA multi-operations scheduling functionality, see the “IP SLAs—Multiple Operation Scheduling” chapter of the *Cisco IOS IP SLAs Configuration Guide*.

IP SLA Operation Threshold Monitoring

To support successful service level agreement monitoring, you must have mechanisms that notify you immediately of any possible violation. IP SLAs can send SNMP traps that are triggered by events such as the following:

- Connection loss
- Timeout
- Round-trip time threshold
- Average jitter threshold
- One-way packet loss
- One-way jitter
- One-way mean opinion score (MOS)
- One-way latency

An IP SLA threshold violation can also trigger another IP SLA operation for further analysis. For example, the frequency could be increased or an Internet Control Message Protocol (ICMP) path echo or ICMP path jitter operation could be initiated for troubleshooting.

ICMP Echo

The ICMP echo operation measures the end-to-end response time between a Cisco device and any other device that uses IP. The response time is computed by measuring the time it takes to send an ICMP echo request message to a destination and receive an ICMP echo reply. Many customers use IP SLA ICMP-based operations, in-house ping testing, or ping-based dedicated probes to measure this response time. The IP SLA ICMP echo operation conforms to the same specifications as ICMP ping testing, and both methods result in the same response times.

UDP Jitter

Jitter is a simple term that describes interpacket delay variance. When multiple packets are sent consecutively at an interval of 10 ms from source to destination, the destination should receive them 10 ms apart (if the network is behaving correctly). However, if there are delays in the network (such as queuing, arriving through alternate routes, and so on), the time interval between packet arrivals might be more or less than 10 ms. A positive jitter value indicates that the packets arrived more than 10 ms apart. A negative jitter value indicates that the packets arrived less than 10 ms apart. If the packets arrive 12 ms apart, the positive jitter is 2 ms; if the packets arrive 8 ms apart, the negative jitter is 2 ms. For delay-sensitive networks, positive jitter values are undesirable, and a jitter value of 0 is ideal.

In addition to monitoring jitter, the IP SLA UDP jitter operation can be used as a multipurpose data gathering operation. The packets generated by IP SLAs carry sequence information and time stamps from the source and operational target that include packet sending and receiving data. Based on this data, UDP jitter operations measure the following:

- Per-direction jitter (source to destination and destination to source)
- Per-direction packet-loss
- Per-direction delay (one-way delay)
- Round-trip delay (average round-trip time)

Because the paths for the sending and receiving of data can be different (asymmetric), you can use the per-direction data to more readily identify where congestion or other problems are occurring in the network.

The UDP jitter operation generates synthetic (simulated) UDP traffic and sends a number of UDP packets, each of a specified size, sent a specified number of milliseconds apart, from a source router to a target router, at a given frequency. By default, ten packet-frames, each with a payload size of 10 bytes are generated every 10 ms, and the operation is repeated every 60 seconds. You can configure each of these parameters to best simulate the IP service you want to provide.

To provide accurate one-way delay (latency) measurements, time synchronization (as provided by NTP) is required between the source and the target device. Time synchronization is not required for the one-way jitter and packet loss measurements. If the time is not synchronized between the source and target devices, one-way jitter and packet loss data is returned, but values of 0 are returned for the one-way delay measurements provided by the UDP jitter operation.

How to Configure IP SLAs Operations

This section does not include configuration information for all available operations as the configuration information details are included in the *Cisco IOS IP SLAs Configuration Guide*. It does include several operations as examples, including configuring the responder, configuring a UDP jitter operation, which requires a responder, and configuring an ICMP echo operation, which does not require a responder. For details about configuring other operations, see the *Cisco IOS IP SLAs Configuration Guide*.

Default Configuration

No IP SLAs operations are configured.

Configuration Guidelines

For information on the IP SLA commands, see the *Cisco IOS IP SLAs Command Reference, Release 12.4T* command reference.

For detailed descriptions and configuration procedures, see the *Cisco IOS IP SLAs Configuration Guide, Release 12.4TL*.

Not all of the IP SLA commands or operations described in the referenced guide are supported on the device. The device supports IP service level analysis by using UDP jitter, UDP echo, HTTP, TCP connect, ICMP echo, ICMP path echo, ICMP path jitter, FTP, DNS, and DHCP, as well as multiple operation scheduling and proactive threshold monitoring. It does not support VoIP service levels using the gatekeeper registration delay operations measurements.

Before configuring any IP SLAs application, you can use the **show ip sla application** privileged EXEC command to verify that the operation type is supported on your software image. This is an example of the output from the command:

```
Switch# show ip sla application

      IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
  icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
  dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
  IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured      : 0
Number of active Entries          : 0
Number of pending Entries         : 0
Number of inactive Entries        : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

Configuring the IP SLA Responder

The IP SLA responder is available only on Cisco IOS software-based devices, including some Layer 2 devices that do not support full IP SLA functionality.

Follow these steps to configure the IP SLA responder on the target device (the operational target):

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla responder {tcp-connect | udp-echo} ipaddress ip-address port port-number**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip sla responder {tcp-connect udp-echo} ipaddress ip-address port port-number Example: <pre>Switch(config)# ip sla responder udp-echo 172.29.139.134 5000</pre>	Configures the device as an IP SLA responder. The keywords have these meanings: <ul style="list-style-type: none"> • tcp-connect—Enables the responder for TCP connect operations. • udp-echo—Enables the responder for User Datagram Protocol (UDP) echo or jitter operations. • ipaddress ip-address—Enter the destination IP address. • port port-number—Enter the destination port number. <p>Note The IP address and port number must match those configured on the source device for the IP SLA operation.</p>
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Implementing IP SLA Network Performance Measurement

Follow these steps to implement IP SLA network performance measurement on your device:

Before you begin

Use the **show ip sla application** privileged EXEC command to verify that the desired operation type is supported on your software image.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla operation-number**
4. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
5. **frequency** *seconds*
6. **threshold** *milliseconds*
7. **exit**
8. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*]} | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip sla operation-number Example: <pre>Switch(config)# ip sla 10</pre>	Creates an IP SLA operation, and enters IP SLA configuration mode.
Step 4	udp-jitter { <i>destination-ip-address</i> <i>destination-hostname</i> } <i>destination-port</i> [source-ip { <i>ip-address</i> <i>hostname</i> }]	Configures the IP SLA operation as the operation type of your choice (a UDP jitter operation is used in the example),

	Command or Action	Purpose
	<p>[source-port <i>port-number</i>] [control {enable disable}] [num-packets <i>number-of-packets</i>] [interval <i>interpacket-interval</i>]</p> <p>Example:</p> <pre>Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000</pre>	<p>and enters its configuration mode (UDP jitter configuration mode is used in the example).</p> <ul style="list-style-type: none"> • <i>destination-ip-address</i> <i>destination-hostname</i>—Specifies the destination IP address or hostname. • <i>destination-port</i>—Specifies the destination port number in the range from 1 to 65535. • (Optional) source-ip {<i>ip-address</i> <i>hostname</i>}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLA chooses the IP address nearest to the destination • (Optional) source-port <i>port-number</i>—Specifies the source port number in the range from 1 to 65535. When a port number is not specified, IP SLA chooses an available port. • (Optional) control—Enables or disables sending of IP SLA control messages to the IP SLA responder. By default, IP SLA control messages are sent to the destination device to establish a connection with the IP SLA responder • (Optional) num-packets <i>number-of-packets</i>—Enters the number of packets to be generated. The range is 1 to 6000; the default is 10. • (Optional) interval <i>inter-packet-interval</i>—Enters the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
<p>Step 5</p>	<p>frequency <i>seconds</i></p> <p>Example:</p> <pre>Switch(config-ip-sla-jitter)# frequency 45</pre>	<p>(Optional) Configures options for the SLA operation. This example sets the rate at which a specified IP SLA operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.</p>
<p>Step 6</p>	<p>threshold <i>milliseconds</i></p> <p>Example:</p> <pre>Switch(config-ip-sla-jitter)# threshold 200</pre>	<p>(Optional) Configures threshold conditions. This example sets the threshold of the specified IP SLA operation to 200. The range is from 0 to 60000 milliseconds.</p>
<p>Step 7</p>	<p>exit</p> <p>Example:</p> <pre>Switch(config-ip-sla-jitter)# exit</pre>	<p>Exits the SLA operation configuration mode (UDP jitter configuration mode in this example), and returns to global configuration mode.</p>

	Command or Action	Purpose
Step 8	<p>ip sla schedule <i>operation-number</i> [life {forever <i>seconds</i>}] [start-time {<i>hh:mm</i> [:<i>ss</i>] [<i>month day</i> <i>day month</i>]}] [pending now after <i>hh:mm:ss</i>] [ageout <i>seconds</i>] [recurring]</p> <p>Example:</p> <pre>Switch(config)# ip sla schedule 10 start-time now life forever</pre>	<p>Configures the scheduling parameters for an individual IP SLA operation.</p> <ul style="list-style-type: none"> • <i>operation-number</i>—Enter the RTR entry number. • (Optional) life—Sets the operation to run indefinitely (forever) or for a specific number of <i>seconds</i>. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). • (Optional) start-time—Enters the time for the operation to begin collecting information: <ul style="list-style-type: none"> To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter pending to select no information collection until a start time is selected. Enter now to start the operation immediately. Enter after <i>hh:mm:ss</i> to show that the operation should start after the entered time has elapsed. • (Optional) ageout <i>seconds</i>—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). • (Optional) recurring—Set the operation to automatically run every day.
Step 9	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

UDP Jitter Configuration

This example shows how to configure a UDP jitter IP SLA operation:

```
Switch(config)# ip sla 10
Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000
Switch(config-ip-sla-jitter)# frequency 30
Switch(config-ip-sla-jitter)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 10.0.0.10/10.0.0.1
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
  Operation frequency (seconds): 30
  Next Scheduled Start Time: Pending trigger
  Group Scheduled : FALSE
  Randomly Scheduled : FALSE
  Life (seconds): 3600
  Entry Ageout (seconds): never
  Recurring (Starting Everyday): FALSE
  Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
  Number of statistic hours kept: 2
  Number of statistic distribution buckets kept: 1
  Statistic distribution interval (milliseconds): 20
Enhanced History:
```

Analyzing IP Service Levels by Using the UDP Jitter Operation

Follow these steps to configure a UDP jitter operation on the source device:

Before you begin

You must enable the IP SLA responder on the target device (the operational target) to configure a UDP jitter operation on the source device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla operation-number**

4. **udp-jitter** *{destination-ip-address | destination-hostname}* *destination-port* [**source-ip** *{ip-address | hostname}*] [**source-port** *port-number*] [**control** *{enable | disable}*] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
5. **frequency** *seconds*
6. **exit**
7. **ip sla schedule** *operation-number* [**life** *{forever | seconds}*] [**start-time** *{hh:mm [:ss] [month day | day month] | pending | now | after hh:mm:ss}*] [**ageout** *seconds*] [**recurring**]
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip sla operation-number Example: Device(config)# ip sla 10	Creates an IP SLA operation, and enters IP SLA configuration mode.
Step 4	udp-jitter <i>{destination-ip-address destination-hostname}</i> <i>destination-port</i> [source-ip <i>{ip-address hostname}</i>] [source-port <i>port-number</i>] [control <i>{enable disable}</i>] [num-packets <i>number-of-packets</i>] [interval <i>interpacket-interval</i>] Example: Device(config-ip-sla)# udp-jitter 172.29.139.134 5000 source-ip 172.29.139.140 source-port 4000	Configures the IP SLA operation as a UDP jitter operation, and enters UDP jitter configuration mode. <ul style="list-style-type: none"> • <i>destination-ip-address destination-hostname</i>: Specifies the destination IP address or hostname. • <i>destination-port</i>: Specifies the destination port number in the range from 1 to 65535. • (Optional) source-ip <i>{ip-address hostname}</i>: Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLA chooses the IP address nearest to the destination. • (Optional) source-port <i>port-number</i>: Specifies the source port number in the range from 1 to 65535. When a port number is not specified, IP SLA chooses an available port.

	Command or Action	Purpose
		<p>Note If the udp-jitter command does not have the source port configured, UDP chooses any random port for control packets. In case UDP chooses the reserved port 1967, it may result in high CPU utilisation by the IP SLA responder.</p> <ul style="list-style-type: none"> • (Optional) control: Enables or disables sending of IP SLA control messages to the IP SLA responder. By default, IP SLA control messages are sent to the destination device to establish a connection with the IP SLA responder. • (Optional) num-packets <i>number-of-packets</i>: Enters the number of packets to be generated. The range is 1 to 6000; the default is 10. • (Optional) interval <i>inter-packet-interval</i>: Enters the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
Step 5	<p>frequency <i>seconds</i></p> <p>Example:</p> <pre>Device(config-ip-sla-jitter)# frequency 45</pre>	(Optional) Sets the rate at which a specified IP SLA operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 6	<p>exit</p> <p>Example:</p> <pre>Device(config-ip-sla-jitter)# exit</pre>	Exits UDP jitter configuration mode, and returns to global configuration mode.
Step 7	<p>ip sla schedule <i>operation-number</i> [life {forever <i>seconds</i>}] [start-time {<i>hh:mm</i> [:<i>ss</i>] [<i>month day</i> <i>day month</i>] pending now after <i>hh:mm:ss</i>] [ageout <i>seconds</i>] [recurring]</p> <p>Example:</p> <pre>Device(config)# ip sla schedule 10 start-time now life forever</pre>	<p>Configures the scheduling parameters for an individual IP SLA operation.</p> <ul style="list-style-type: none"> • <i>operation-number</i>: Enter the RTR entry number. • (Optional) life: Sets the operation to run indefinitely (forever) or for a specific number of <i>seconds</i>. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). • (Optional) start-time: Enters the time for the operation to begin collecting information: To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month.

	Command or Action	Purpose
		<p>Enter pending to select no information collection until a start time is selected.</p> <p>Enter now to start the operation immediately.</p> <p>Enter after <i>hh:mm:ss</i> to show that the operation should start after the entered time has elapsed.</p> <ul style="list-style-type: none"> • (Optional) ageout <i>seconds</i>: Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). • (Optional) recurring: Set the operation to automatically run every day.
Step 8	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 10	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a UDP Jitter IP SLA Operation

This example shows how to configure a UDP jitter IP SLA operation:

```
Device(config)# ip sla 10
Device(config-ip-sla)# udp-jitter 172.29.139.134 5000 source-ip 172.29.139.140 source-port
4000
Device(config-ip-sla-jitter)# frequency 30
Device(config-ip-sla-jitter)# exit
Device(config)# ip sla schedule 10 start-time now life forever
Device(config)# end
Device# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
```

```

Type of operation to perform: udp-jitter
Target address/Source address: 10.0.0.10/10.0.0.1
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
  Operation frequency (seconds): 30
  Next Scheduled Start Time: Pending trigger
  Group Scheduled : FALSE
  Randomly Scheduled : FALSE
  Life (seconds): 3600
  Entry Ageout (seconds): never
  Recurring (Starting Everyday): FALSE
  Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
  Number of statistic hours kept: 2
  Number of statistic distribution buckets kept: 1
  Statistic distribution interval (milliseconds): 20
Enhanced History:

```

Analyzing IP Service Levels by Using the ICMP Echo Operation

Follow these steps to configure an ICMP echo operation on the source device:

Before you begin

This operation does not require the IP SLA responder to be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla operation-number**
4. **icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*]
5. **frequency** *seconds*
6. **exit**
7. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch> enable</pre>	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>ip sla operation-number</p> <p>Example:</p> <pre>Switch(config)# ip sla 10</pre>	Creates an IP SLA operation and enters IP SLA configuration mode.
Step 4	<p>icmp-echo {<i>destination-ip-address</i> <i>destination-hostname</i>} [source-ip {<i>ip-address</i> <i>hostname</i>} source-interface <i>interface-id</i>]</p> <p>Example:</p> <pre>Switch(config-ip-sla)# icmp-echo 172.29.139.134</pre>	<p>Configures the IP SLA operation as an ICMP Echo operation and enters ICMP echo configuration mode.</p> <ul style="list-style-type: none"> • <i>destination-ip-address</i> <i>destination-hostname</i>—Specifies the destination IP address or hostname. • (Optional) source-ip {<i>ip-address</i> <i>hostname</i>}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLA chooses the IP address nearest to the destination. • (Optional) source-interface <i>interface-id</i>—Specifies the source interface for the operation.
Step 5	<p>frequency <i>seconds</i></p> <p>Example:</p> <pre>Switch(config-ip-sla-echo)# frequency 30</pre>	(Optional) Sets the rate at which a specified IP SLA operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 6	<p>exit</p> <p>Example:</p> <pre>Switch(config-ip-sla-echo)# exit</pre>	Exits UDP echo configuration mode, and returns to global configuration mode.
Step 7	<p>ip sla schedule <i>operation-number</i> [life {forever <i>seconds</i>}] [start-time {<i>hh:mm</i> [:<i>ss</i>] [<i>month</i> <i>day</i> <i>day</i> <i>month</i>] pending now after <i>hh:mm:ss</i>}] [ageout <i>seconds</i>] [recurring]</p> <p>Example:</p>	<p>Configures the scheduling parameters for an individual IP SLA operation.</p> <ul style="list-style-type: none"> • <i>operation-number</i>—Enter the RTR entry number. • (Optional) life—Sets the operation to run indefinitely (forever) or for a specific number of <i>seconds</i>. The

	Command or Action	Purpose
	<pre>Switch(config)# ip sla schedule 5 start-time now life forever</pre>	<p>range is from 0 to 2147483647. The default is 3600 seconds (1 hour)</p> <ul style="list-style-type: none"> (Optional) start-time—Enter the time for the operation to begin collecting information: To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter pending to select no information collection until a start time is selected. Enter now to start the operation immediately. Enter after <i>hh:mm:ss</i> to indicate that the operation should start after the entered time has elapsed. (Optional) ageout <i>seconds</i>—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds; the default is 0 seconds (never ages out). (Optional) recurring—Sets the operation to automatically run every day.
Step 8	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 10	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring an ICMP Echo IP SLA Operation

This example shows how to configure an ICMP echo IP SLA operation:

```
Switch(config)# ip sla 12
```

```

Switch(config-ip-sla)# icmp-echo 172.29.139.134
Switch(config-ip-sla-echo)# frequency 30
Switch(config-ip-sla-echo)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 22
IP SLAs, Infrastructure Engine-II.

Entry number: 12
Owner:
Tag:
Type of operation to perform: echo
Target address: 2.2.2.2
Source address: 0.0.0.0
Request size (ARR data portion): 28
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Schedule:
  Operation frequency (seconds): 60
  Next Scheduled Start Time: Pending trigger
  Group Scheduled : FALSE
  Randomly Scheduled : FALSE
  Life (seconds): 3600
  Entry Ageout (seconds): never
  Recurring (Starting Everyday): FALSE
  Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
  Number of statistic hours kept: 2
  Number of statistic distribution buckets kept: 1
  Statistic distribution interval (milliseconds): 20
History Statistics:
  Number of history Lives kept: 0
  Number of history Buckets kept: 15
  History Filter Type: None
Enhanced History:

```

Monitoring IP SLA Operations

The following table describes the commands used to display IP SLA operation configurations and results:

Table 61: Monitoring IP SLA Operations

show ip sla application	Displays global information.
show ip sla authentication	Displays IP SLA authentication.
show ip sla configuration [<i>entry-number</i>]	Displays configuration values for all IP SLA operations or a specific operation.
show ip sla enhanced-history { <i>collection-statistics</i> <i>distribution statistics</i> } [<i>entry-number</i>]	Displays enhanced history or distribution statistics for a specific IP SLA operation.
show ip sla ethernet-monitor configuration [<i>entry-number</i>]	Displays IP SLA automatic configuration.

show ip sla group schedule [<i>schedule-entry-number</i>]	Displays IP SLA group sched
show ip sla history [<i>entry-number</i> full tabular]	Displays history collected for
show ip sla mpls-lsp-monitor { collection-statistics configuration ldp operational-state scan-queue summary [<i>entry-number</i>] neighbors }	Displays MPLS label switche operations.
show ip sla reaction-configuration [<i>entry-number</i>]	Displays the configured proact all IP SLA operations or a spe
show ip sla reaction-trigger [<i>entry-number</i>]	Displays the reaction trigger in or a specific operation.
show ip sla responder	Displays information about th
show ip sla statistics [<i>entry-number</i> aggregated details]	Displays current or aggregate

Monitoring IP SLA Operation Examples

The following example shows all IP SLAs by application:

```
Switch# show ip sla application

      IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
      icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
      dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
      IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured      : 0
Number of active Entries          : 0
Number of pending Entries         : 0
Number of inactive Entries        : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

The following example shows all IP SLA distribution statistics:

```
Switch# show ip sla enhanced-history distribution-statistics

Point by point Enhanced History
Entry      = Entry Number
Int        = Aggregation Interval
BucI       = Bucket Index
StartT     = Aggregation Start Time
Pth        = Path index
Hop        = Hop in path index
Comps      = Operations completed
OvrTh      = Operations completed over thresholds
SumCmp     = Sum of RTT (milliseconds)
SumCmp2L   = Sum of RTT squared low 32 bits (milliseconds)
```



```
SumCmp2H = Sum of RTT squared high 32 bits (milliseconds)
TMax     = RTT maximum (milliseconds)
TMin     = RTT minimum (milliseconds)
```

```
Entry Int BucI StartT      Pth Hop Comps OvrTh SumCmp      SumCmp2L   SumCmp2H   T
Max    TMin
```




CHAPTER 35

Configuring Enhanced Object Tracking

- [Finding Feature Information](#), on page 659
- [Information About Enhanced Object Tracking](#), on page 659
- [How to Configure Enhanced Object Tracking](#), on page 661
- [Monitoring Enhanced Object Tracking](#), on page 674

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Information About Enhanced Object Tracking

Enhanced Object Tracking Overview

Before the introduction of the Enhanced Object Tracking feature, Hot Standby Router Protocol (HSRP) had a simple tracking mechanism that allowed you to track the interface line-protocol state only. If the line-protocol state of the interface went down, the HSRP priority of the router was reduced, allowing another HSRP router with a higher priority to become active.

The Enhanced Object Tracking feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by processes other than HSRP. This feature allows the tracking of other objects in addition to the interface line-protocol state.

A client process such as HSRP, Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP), can register its interest in tracking objects and then be notified when the tracked object changes state.

Each tracked object has a unique number that is specified in the tracking command-line interface (CLI). Client processes use this number to track a specific object. The tracking process periodically polls the tracked object for value changes and sends any changes (as up or down values) to interested client processes, either

immediately or after a specified delay. Several clients can track the same object, and can take different actions when the object changes state.

You can also track a combination of objects in a list by using either a weight threshold or a percentage threshold to measure the state of the list. You can combine objects using Boolean logic. A tracked list with a Boolean “AND” function requires that each object in the list be in an up state for the tracked object to be up. A tracked list with a Boolean “OR” function needs only one object in the list to be in the up state for the tracked object to be up.

Tracking Interface Line-Protocol or IP Routing State

You can track either the interface line protocol state or the interface IP routing state. When you track the IP routing state, these three conditions are required for the object to be up:

- IP routing must be enabled and active on the interface.
- The interface line-protocol state must be up.
- The interface IP address must be known.

If all three of these conditions are not met, the IP routing state is down.

Tracked Lists

You can configure a tracked list of objects with a Boolean expression, a weight threshold, or a percentage threshold. A tracked list contains one or more objects. An object must exist before it can be added to the tracked list.

- You configure a Boolean expression to specify calculation by using either “AND” or “OR” operators.
- When you measure the tracked list state by a weight threshold, you assign a weight number to each object in the tracked list. The state of the tracked list is determined by whether or not the threshold was met. The state of each object is determined by comparing the total weight of all objects against a threshold weight for each object.
- When you measure the tracked list by a percentage threshold, you assign a percentage threshold to all objects in the tracked list. The state of each object is determined by comparing the assigned percentages of each object to the list.

Tracking Other Characteristics

You can also use the enhanced object tracking for tracking other characteristics.

- You can track the reachability of an IP route by using the **track ip route reachability** global configuration command.
- You can use the **track ip route metric threshold** global configuration command to determine if a route is above or below threshold.
- You can use the **track resolution** global configuration command to change the metric resolution default values for routing protocols.
- You can use the **track timer tracking** configuration command to configure the tracking process to periodically poll tracked objects.

Use the **show track** privileged EXEC command to verify enhanced object tracking configuration.

IP SLAs Object Tracking

Cisco IOS IP Service Level Agreements (IP SLAs) is a network performance measurement and diagnostics tool that uses active monitoring by generating traffic to measure network performance. Cisco IP SLAs operations collect real-time metrics that you can use for network troubleshooting, design, and analysis.

Object tracking of IP SLAs operations allows clients to track the output from IP SLAs objects and use this information to trigger an action. Every IP SLAs operation maintains an SNMP operation return-code value, such as OK or OverThreshold, that can be interpreted by the tracking process. You can track two aspects of IP SLAs operation: state and reachability. For state, if the return code is OK, the track state is up; if the return code is not OK, the track state is down. For reachability, if the return code is OK or OverThreshold, reachability is up; if not OK, reachability is down.

Static Route Object Tracking

Static routing support using enhanced object tracking provides the ability for the device to use ICMP pings to identify when a pre-configured static route or a DHCP route goes down. When tracking is enabled, the system tracks the state of the route and informs the client when that state changes. Static route object tracking uses Cisco IP SLAs to generate ICMP pings to monitor the state of the connection to the primary gateway.

How to Configure Enhanced Object Tracking

Configuring Tracking for Line State Protocol or IP Routing State on an Interface

Follow these steps to track the line-protocol state or IP routing state of an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track *object-number* interface *interface-id* line-protocol**
4. **delay { *object-number* upseconds[downseconds][upseconds]downseconds }**
5. **exit**
6. **track *object-number* interface *interface-id* ip routing**
7. **delay { *object-number* upseconds[downseconds][upseconds]downseconds }**
8. **end**
9. **show track *object-number***

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	track object-number interface interface-id line-protocol Example: Switch(config)# track 33 interface gigabitethernet 1/0/1 line-protocol	(Optional) Creates a tracking list to track the line-protocol state of an interface and enter tracking configuration mode. <ul style="list-style-type: none"> • The object-number identifies the tracked object and can be from 1 to 500. • The interface interface-id is the interface being tracked.
Step 4	delay { <i>object-number</i> upseconds [downseconds] [upseconds] downseconds }	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 5	exit	Returns to global configuration mode.
Step 6	track object-number interface interface-id ip routing Example: Switch(config)# track 33 interface gigabitethernet 1/0/1 ip routing	(Optional) Creates a tracking list to track the IP routing state of an interface and enter tracking configuration mode. IP route tracking tracks an IP route in the routing table and the ability of an interface to route IP packets. <ul style="list-style-type: none"> • The object-number identifies the tracked object and can be from 1 to 500. • The interface interface-id is the interface being tracked.
Step 7	delay { <i>object-number</i> upseconds [downseconds] [upseconds] downseconds }	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 8	end	Returns to privileged EXEC mode.
Step 9	show track object-number	Verifies that the specified objects are being tracked.

Configuring Tracked Lists

Configuring a Tracked List with a Weight Threshold

To track by weight threshold, configure a tracked list of objects, specify that weight is used as the threshold, and configure a weight for each of its objects. The state of each object is determined by comparing the total weight of all objects that are up against a threshold weight for each object.

You cannot use the Boolean “NOT” operator in a weight threshold list.

Follow these steps to configure a tracked list of objects by using a weight threshold and to configure a weight for each object:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-numberlist* **threshold** {**weight**}
4. **object** *object-number*[**weightweight-number**]
5. **threshold weight** {**upnumber**[[**downnumber**]]}
6. **delay** { **upseconds**[**downseconds**][[**upseconds**]**downseconds**}
7. **end**
8. **show track***object-number*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	track <i>track-numberlist</i> threshold { weight } Example: <pre>Switch(config)# track 4 list threshold weight</pre>	Configures a tracked list object, and enters tracking configuration mode. The track-number can be from 1 to 500. <ul style="list-style-type: none"> • threshold—Specifies the state of the tracked list based on a threshold. • weight— Specifies that the threshold is based on weight.
Step 4	object <i>object-number</i> [weightweight-number] Example: <pre>Switch(config)# object 2 weight 15</pre>	Specifies the object to be tracked. The range is from 1 to 500. The optional weightweight-number specifies the threshold weight for the object. The range is from 1 to 255. <p>Note An object must exist before you can add it to a tracked list.</p>

	Command or Action	Purpose
Step 5	threshold weight { <i>upnumber</i> [[<i>downnumber</i>]]} Example: <pre>Switch(config-track)# threshold weight up 30 down 10</pre>	(Optional) Specifies the threshold weight. <ul style="list-style-type: none"> • <i>upnumber</i>—The range is from 1 to 255. • <i>downnumber</i>—(Optional)The range depends on the number selected for the <i>upnumber</i>. If you configure the <i>upnumber</i> as 25, the range shown for the down number is 0 to 24.
Step 6	delay { <i>upseconds</i> [[<i>downseconds</i>]][<i>upseconds</i>] <i>downseconds</i> }	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 7	end	Returns to privileged EXEC mode.
Step 8	show track <i>object-number</i>	Verify that the specified objects are being tracked.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Tracked List with a Percentage Threshold

To track by percentage threshold, configure a tracked list of objects, specify that a percentage will be used as the threshold, and specify a percentage for all objects in the list. The state of the list is determined by comparing the assigned percentage of each object to the list.

You cannot use the Boolean “NOT” operator in a percentage threshold list.

Follow these steps to configure a tracked list of objects by using a percentage threshold:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-number***list threshold** {*percentage*}
4. **object** *object-number*
5. **threshold percentage** {*upnumber*[[*downnumber*]]}
6. **delay** { *upseconds*[[*downseconds*]][*upseconds*]*downseconds*}
7. **end**
8. **show track***object-number*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: <pre>Switch> enable</pre>	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	track track-numberlist threshold {percentage} Example: <pre>Switch(config)# track 4 list threshold percentage</pre>	Configures a tracked list object, and enters tracking configuration mode. The track-number can be from 1 to 500. <ul style="list-style-type: none"> • threshold—Specifies the state of the tracked list based on a threshold. • percentage— Specifies that the threshold is based on percentage.
Step 4	object object-number Example: <pre>Switch(config)# object 1</pre>	Specifies the object to be tracked. The range is from 1 to 500. Note An object must exist before you can add it to a tracked list.
Step 5	threshold percentage {upnumber}[downnumber]} Example: <pre>Switch(config)# threshold percentage up 51 down 10</pre>	(Optional) Specifies the threshold percentage. <ul style="list-style-type: none"> • upnumber—The range is from 1 to 100. • downnumber—(Optional)The range depends on the number selected for the upnumber. If you configure the upnumber as 25, the range shown for the down number is 0 to 24.
Step 6	delay {upseconds[downseconds][upseconds]downseconds}	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 7	end	Returns to privileged EXEC mode.
Step 8	show trackobject-number	Verify that the specified objects are being tracked.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring HSRP Object Tracking

Follow these steps to configure a standby HSRP group to track an object and change the HSRP priority based on the object state:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *object-number* {**interface** *interface-id* {**line-protocol**|**ip routing**}|**ip route** *ip address/prefix-length* {**metric** **threshold**|**reachability**} **list** {**boolean** {**and**|**or**}|} {**threshold** {**weight**|**percentage**}}
4. **exit**
5. **interface** { *interface-id*
6. **standby**[*group-number*]**ip**[*ip-address*]**secondary**]]
7. **standby**[*group-number*]**track**[*object-number*]**decrement** *priority-decrement*]]
8. **end**
9. **show standby**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	track <i>object-number</i> { interface <i>interface-id</i> { line-protocol ip routing } ip route <i>ip address/prefix-length</i> { metric threshold reachability } list { boolean { and or } } { threshold { weight percentage }}	(Optional) Create a tracking list to track the configured state and enter tracking configuration mode. <ul style="list-style-type: none"> • The <i>object-number</i> identifies the tracked object and can be from 1 to 500. • Enter interface <i>interface-id</i> to select an interface to track. • Enter line-protocol to track the interface line protocol state or enter ip routing to track the interface IP routing state . • Enter ip route <i>ip-address/prefix-length</i> to track the state of an IP route.

	Command or Action	Purpose
		<ul style="list-style-type: none"> Enter metric threshold to track the threshold metric or enter reachability to track if the route is reachable. <p>The default up threshold is 254 and the default down threshold is 255.</p> <ul style="list-style-type: none"> Enter list to track objects grouped in a list. <p>Note Repeat this step for each interface to be tracked.</p>
Step 4	exit	Return to global configuration mode.
Step 5	interface { <i>interface-id</i>	Enter interface configuration mode.
Step 6	standby [<i>group-number</i>] ip [<i>ip-address</i> secondary]	<p>Creates (or enables) the HSRP group by using its number and virtual IP address.</p> <ul style="list-style-type: none"> (Optional) <i>group-number</i>—Enters a group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. (Optional on all but one interface) <i>ip-address</i>—Specifies the virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. (Optional) secondary—Specifies that the IP address is a secondary hot standby router interface. If this keyword is omitted, the configured address is the primary IP address.
Step 7	standby [<i>group-number</i>] track [<i>object-number</i> [decrement <i>priority-decrement</i>]]	<p>Configures HSRP to track an object and change the hot standby priority based on the state of the object.</p> <ul style="list-style-type: none"> (Optional) <i>group-number</i>—Enters the group number to which the tracking applies. <i>object-number</i>—Enters a number representing the object to be tracked. The range is from 1 to 500; the default is 1. (Optional) secondary—Specifies that the IP address is a secondary hot standby router interface. If this keyword is omitted, the configured address is the primary IP address. (Optional)decrement<i>priority-decrement</i>—Specifies the amount by which the hot standby priority for the router is decremented (or incremented) when the

	Command or Action	Purpose
		tracked object goes down (or comes back up). The range is from 1 to 255; the default is 10.
Step 8	<code>end</code>	Returns to privileged EXEC mode.
Step 9	<code>show standby</code>	Verifies the standby router IP address and tracking states.
Step 10	<code>copy running-config startup-config</code> Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring IP SLAs Object Tracking

Follow these steps to track the state of an IP SLAs operation or the reachability of an IP SLAs IP host:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `track object-number ip sla operation-number {state | reachability}`
4. `delay { upseconds[downseconds][upseconds]downseconds}`
5. `end`
6. `show trackobject-number`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code> Example: Switch> <code>enable</code>	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	<code>configure terminal</code> Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>track object-number ip sla operation-number {state reachability}</code> Example: Switch(config)# <code>track 2 ip sla 123 state</code>	Enters tracking configuration mode to track the state of an IP SLAs operation. • <i>object-number</i> range is from 1 to 500. • <i>operation-number</i> range is from 1 to 2147483647.

	Command or Action	Purpose
Step 4	<code>delay { upseconds[downseconds][upseconds]downseconds}</code>	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 5	<code>end</code>	Returns to privileged EXEC mode.
Step 6	<code>show trackobject-number</code>	Verifies that the specified objects are being tracked.
Step 7	<code>copy running-config startup-config</code> Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Static Route Object Tracking

Configuring a Primary Interface for Static Routing

Follow these steps to configure a primary interface for static routing:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interfaceinterface-id`
4. `descriptionstring`
5. `ip addressip-address mask[secondary]`
6. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code> Example: <code>Switch> enable</code>	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	<code>configure terminal</code> Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 3	<code>interfaceinterface-id</code>	Selects a primary or secondary interface and enters interface configuration mode.

	Command or Action	Purpose
Step 4	<code>descriptionstring</code>	Adds a description to the interface.
Step 5	<code>ip addressip-address mask[secondary]</code>	Sets the primary or secondary IP address for the interface.
Step 6	<code>exit</code>	Returns to global configuration mode.

Configuring a Primary Interface for DHCP

Follow these steps to configure a primary interface for DHCP:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interfaceinterface-id`
4. `descriptionstring`
5. `ip dhcp client route tracknumber`
6. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<code>interfaceinterface-id</code>	Selects a primary or secondary interface and enters interface configuration mode.
Step 4	<code>descriptionstring</code>	Adds a description to the interface.
Step 5	<code>ip dhcp client route tracknumber</code>	Configures the DHCP client to associate any added routes with the specified track number. Valid numbers are from 1 to 500.
Step 6	<code>exit</code>	Returns to global configuration mode.

Configuring IP SLAs Monitoring Agent

You can configure an IP SLAs agent to ping an IP address using a primary interface and a track object to monitor the state of the agent.

Follow these steps to configure network monitoring with Cisco IP SLAs:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla***operation number*
4. **icmp-echo** { *destination ip-address|destination hostname* [**source - ipaddr** { *ip-address|hostname* **source-interface** *interface-id*] }
5. **timeout***milliseconds*
6. **frequency***seconds*
7. **threshold***milliseconds*
8. **exit**
9. **ip sla schedule** *operation-number* [**life** { **forever** | *seconds* }] [**start-time** *time* | **pending** | **now** | **after** *time*] [**ageout** *seconds*] [**recurring**]
10. **track** *object-number* **rrtr** *operation-number* **state** *reachability*
11. **end**
12. **show track** *object-number*
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip sla <i>operation number</i>	Begins configuring a Cisco IP SLAs operation and enters IP SLA configuration mode.
Step 4	icmp-echo { <i>destination ip-address destination hostname</i> [source - ipaddr { <i>ip-address hostname</i> source-interface <i>interface-id</i>] }	Configures a Cisco IP SLAs end-to-end ICMP echo response time operation and enter IP SLAs ICMP echo configuration mode.
Step 5	timeout <i>milliseconds</i>	Sets the amount of time for which the operation waits for a response from its request packet.
Step 6	frequency <i>seconds</i>	Sets the rate at which the operation is sent into the network.
Step 7	threshold <i>milliseconds</i>	Sets the rising threshold (hysteresis) that generates a reaction event and stores history information for the operation.

	Command or Action	Purpose
Step 8	<code>exit</code>	Exits IP SLAs ICMP echo configuration mode.
Step 9	<code>ip sla schedule operation-number</code> [<code>life {forever seconds}</code>] [<code>start-time</code>] [<code>pending</code>] [<code>no</code>] [<code>after-time</code>] [<code>egout</code>] [<code>seconds</code>] [<code>recuring</code>] Example: <code>Switch(config)# track 2 200 state</code>	Configures the scheduling parameters for a single IP SLAs operation. <ul style="list-style-type: none"> • <code>object-number</code> range is from 1 to 500. • <code>operation-number</code> range is from 1 to 2147483647.
Step 10	<code>track object-number rtr</code> <code>operation-number state reachability</code>	Tracks the state of a Cisco IOS IP SLAs operation and enter tracking configuration mode.
Step 11	<code>end</code>	Returns to privileged EXEC mode.
Step 12	<code>show track object-number</code>	Verifies that the specified objects are being tracked.
Step 13	<code>copy running-config startup-config</code> Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a Routing Policy and a Default Route

Follow these steps to configure a routing policy for backup static routing by using object tracking.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `access-list access-list-number`
4. `route-map map tag` [`permit|deny`] [`sequence-number`]
5. `match ip address` {`access-list number` [`permit|deny`] [`sequence-number`]}
6. `set ip next-hop dynamic dhcp`
7. `set interface interface-id`
8. `exit`
9. `ip local policy route-map map tag`
10. `ip route prefix mask` {`ip address|interface-id` [`ip address`]} [`distance`] [`name`] [`permanent|track track-number`] [`tag tag`]
11. `end`
12. `show ip route track table`
13. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code> Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	access-list <i>access-list-number</i>	Defines an extended IP access list. Configure any optional characteristics.
Step 4	route-map <i>map tag</i> [permit deny][<i>sequence-number</i>]	Enters route-map configuration mode and define conditions for redistributing routes from one routing protocol to another.
Step 5	match ip address { <i>access-list number</i> [permit deny][<i>sequence-number</i>]	Distribute any routes that have a destination network number address that is permitted by a standard or extended access list or performs policy routing on packets. You can enter multiple numbers or names.
Step 6	set ip next-hop dynamic dhcp	For DHCP networks only. Sets the next hop to the gateway that was most recently learned by the DHCP client.
Step 7	set interface <i>interface-id</i>	For static routing networks only. Indicates where to send output packets that pass a match clause of a route map for policy routing.
Step 8	exit	Returns to global configuration mode.
Step 9	ip local policy route-map <i>map tag</i>	Identifies a route map to use for local policy routing.
Step 10	ip route <i>prefix mask</i> { <i>ip address</i> <i>interface-id</i> [<i>ip address</i>]}[<i>distance</i>][<i>name</i>][permanent][track <i>track-number</i>][<i>tag tag</i>]	For static routing networks only. Establishes static routes. Entering track <i>track-number</i> specifies that the static route is installed only if the configured track object is up.
Step 11	end	Returns to privileged EXEC mode.
Step 12	show ip route track table	Displays information about the IP route track table.
Step 13	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Enhanced Object Tracking

Use the privileged EXEC or user EXEC commands in the table below, to display enhanced object tracking information.

Table 62: Commands for Displaying Tracking Information

Command	Purpose
show ip route track table	Displays information about the IP route tr
show track <i>[object-number]</i>	Displays information about the all tracking
show track brief	Displays VTP status and configuration fo
show track interface <i>[brief]</i>	Displays information about tracked interf
show track ip <i>[object-number]</i> <i>[brief]</i> route	Displays information about tracked IP-ro
show track resolution	Displays the resolution of tracked parame
show track timer	Displays tracked polling interval timers.



CHAPTER 36

Managing Switch Stacks

- [Prerequisites for Switch Stacks, on page 675](#)
- [Restrictions for Switch Stacks, on page 675](#)
- [Information About Switch Stacks, on page 675](#)
- [How to Configure a Switch Stack, on page 685](#)
- [Troubleshooting the Switch Stack, on page 693](#)
- [Monitoring the Device Stack, on page 694](#)
- [Configuration Examples for Switch Stacks, on page 695](#)
- [Additional References for Switch Stacks, on page 698](#)

Prerequisites for Switch Stacks

Restrictions for Switch Stacks

There are no restrictions with Horizontal Stacking on Catalyst 3560cx.

Information About Switch Stacks

Horizontal Stacking

The Catalyst 3560CX series switches supporting 10G SFP+ uplink ports and MGig ports can be part of horizontal stacking. We can use SFP+ with the optical cables and copper cables on the MGig ports to connect boxes placed at different location to form a stack, where the compact boxes are placed in different floors or buildings. We can form half-ring or full-ring based on need, and remaining uplink ports will continue to work as network ports.

When we convert a network port to stack port, it will continue to work as network port without any impact to current running configuration until next reload of switch. All current configurations of that particular network port will be lost after reload of switch once port comes up as Stack port.

When we convert a stack port back to network port it will continue to work as stack port until next reload of switch. After reload port comes up as network port with default configuration.



Note When uplink ports are working as stack ports, those particular uplink interfaces (ex: Te1/0/1) will not be listed in any show command or will not be available under any config command, unlike any other network port. They will be made available only after reload of the switch once ports are converted back to network.

Table 63: C3560CX Switches Supporting Horizontal Stacking

Product ID	Access Ports	Uplinks	Stackable Ports
WS-C3560CX-12PD-S	12 GE	2GE + 2SFP+	2 10G uplink
WS-C3560CX-8XPD-S	6 GE	2multigig + 2SFP+	1 multigig and 1 10G uplink, or 2 multigig, or 2 10G uplink

Switch Stack Membership

A switch stack has up to eight stack members connected through their stack ports. A switch stack always has one active switch.

A standalone device is a device stack with one stack member that also operates as the active switch. You can connect one standalone device to another to create a stack containing two stack members, with one of them as the active switch. You can connect standalone devices to an existing device stack to increase the stack membership.

Figure 61: Creating a Switch Stack from Two Standalone Switches

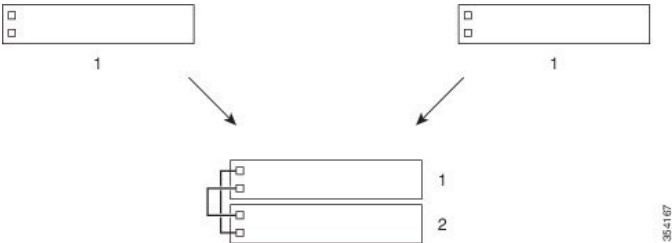
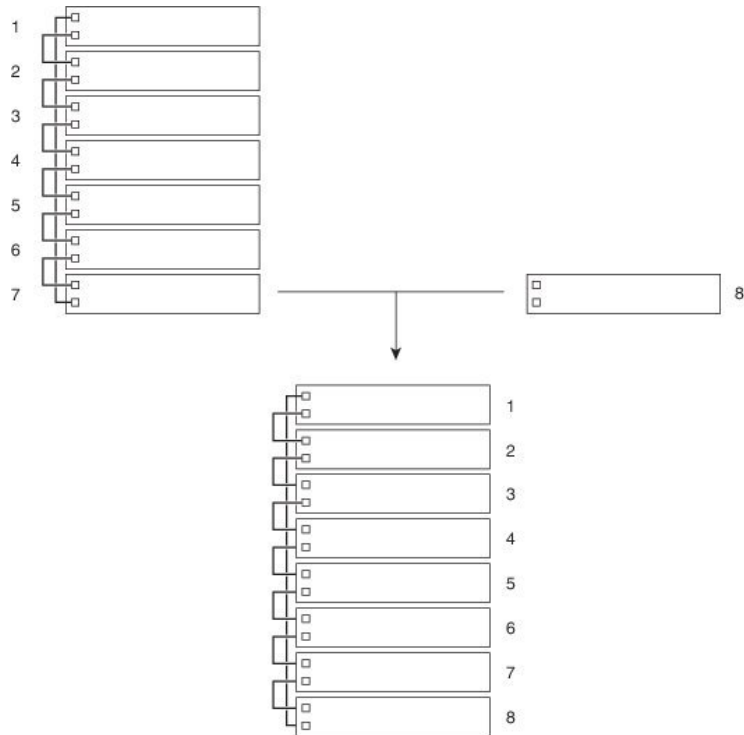


Figure 62: Adding a Standalone Switch to a Switch Stack



354-108

Changes to Switch Stack Membership

If you replace a stack member with an identical model, the new switch functions with exactly the same configuration as the replaced switch, assuming that the new switch (referred to as the provisioned switch) is using the same member number as the replaced switch.

The operation of the switch stack continues uninterrupted during membership changes unless you remove the active switch or you add powered-on standalone switches or switch stacks.

- Adding powered-on switches (merging) causes the active stack of the merging switch stacks to elect an active stack from among themselves. The reelected active stack retains its role and configuration as do its stack members. All remaining switches, including the former active stacks, reload and join the switch stack as stack members. They change their stack member numbers to the lowest available numbers and use the stack configuration of the reelected active stack.
- Removing powered-on stack members causes the switch stack to divide (partition) into two or more switch stacks, each with the same configuration. This can cause:
 - An IP address conflict in your network. If you want the switch stacks to remain separate, change the IP address or addresses of the newly created switch stacks.
 - A MAC address conflict between two members in the stack. You can use the **stack-mac update force** command to resolve the conflict.



Note Make sure that you power off the switches that you add to or remove from the switch stack.

After adding or removing stack members, make sure that the switch stack is operating at full bandwidth . Press the Mode button on a stack member until the Stack mode LED is on. The last two right port LEDs on all switches in the stack should be green. Depending on the switch model, the last two right ports are 10-Gigabit Ethernet ports or small form-factor pluggable (SFP) module ports (10/100/1000 ports). If one or both of these LEDs are not green on any of the switches, the stack is not operating at full bandwidth.

It may take upto 4 seconds for stack convergence when a new stack member is added to the existing switch stack.

If you remove powered-on members but do not want to partition the stack:

- Power off the switches in the newly created switch stacks.
- Reconnect them to the original switch stack through their stack ports.
- Power on the switches.

Stack Member Numbers

The stack member number (1 to 8) identifies each member in the switch stack. The member number also determines the interface-level configuration that a stack member uses. You can display the stack member number by using the **show switch EXEC** command.

A new, out-of-the-box device (one that has not joined a device stack or has not been manually assigned a stack member number) ships with a default stack member number of 1. When it joins a device stack, its default stack member number changes to the lowest available member number in the stack.

Stack members in the same stack cannot have the same stack member number. Every stack member, including a standalone device, retains its member number until you manually change the number or unless the number is already being used by another member in the stack.

- If you manually change the stack member number by using the **switch current-stack-member-number renumber new-stack-member-number** command, the new number goes into effect after that stack member resets (or after you use the **reload slot stack-member-number** privileged EXEC command) and only if that number is not already assigned to any other members in the stack. Another way to change the stack member number is by changing the device_NUMBER environment variable.

If the number is being used by another member in the stack, the device selects the lowest available number in the stack.

If you manually change the number of a stack member and no interface-level configuration is associated with that new member number, that stack member resets to its default configuration.

You cannot use the **switch current-stack-member-number renumber new-stack-member-number** command on a provisioned device. If you do, the command is rejected.

- If you move a stack member to a different device stack, the stack member retains its number only if the number is not being used by another member in the stack. If it is being used, the device selects the lowest available number in the stack.
- If you merge device stacks, the device that join the device stack of a new active device select the lowest available numbers in the stack.

As described in the hardware installation guide, you can use the device port LEDs in Stack mode to visually determine the stack member number of each stack member.

In the **default** mode Stack LED will blink in green color only on the active switch. However, when we scroll the Mode button to **Stack** option - Stack LED will glow green on all the stack members.

When mode button is scrolled to **Stack** option, the switch number of each stack member will be displayed as LEDs on the first five ports of that switch. The switch number is displayed in binary format for all stack members. On the switch, the amber LED indicates value 0 and green LED indicates value 1.

Example for switch number 5 (Binary - 00101):

First five LEDs glow as follows on stack member with switch number 5.

- Port-1 : Amber
- Port-2 : Amber
- Port-3 : Green
- Port-4 : Amber
- Port-5 : Green

Similarly, the first five LEDs glow amber or green, depending on the switch number on all stack members.



Note

- If you connect a Horizontal stack port to a normal network port on other end, stack port transmission/reception will be disabled within 30 seconds if no SDP packets are received from the other end.
- Stack port will not go down but only transmission/reception will be disabled. The log message shown below will be displayed on the console. Once the peer end network port is converted to stack port, transmission/reception on this stack port will be enabled.

```
%STACKMGR-4-HSTACK_LINK_CONFIG: Verify peer stack port setting for hstack
StackPort-1 switch 5 (hostname-switchnumber)
```

Stack Member Priority Values

A higher priority value for a stack member increases the probability of it being elected active switch and retaining its stack member number. The priority value can be 1 to 15. The default priority value is 1. You can display the stack member priority value by using the **show switch EXEC** command.



Note

We recommend assigning the highest priority value to the device that you prefer to be the active device. This ensures that the device is reelected as the active device if a reelection occurs.

To change the priority value for a stack member, use the **switch stack-member-number priority new priority-value** command. For more information, see the “Setting the Stack Member Priority Value” section.

The new priority value takes effect immediately but does not affect the current active device. The new priority value helps determine which stack member is elected as the new active device when the current active device or the device stack resets.

Switch Stack Bridge ID and MAC Address

The MAC address of the active switch determines the stack MAC address.

When the stack initializes, the MAC address of the active switch determines the bridge ID that identifies the stack in the network.

If the active switch changes, the MAC address of the new active switch determines the new bridge ID and stack MAC address.

If the entire switch stack reloads, the switch stack uses the MAC address of the active switch.

Persistent MAC Address on the Switch Stack

You can also configure stack MAC persistency so that the stack MAC address never changes to the new active switch MAC address.

Active and Standby Switch Election and Reelection

The active switch is elected or reelected based on one of these factors and in the order listed:

1. The switch that is currently the active switch.
2. The switch with the highest stack member priority value.



Note We recommend assigning the highest priority value to the switch that you prefer to be the active switch. This ensures that the switch is reelected as active switch if a reelection occurs.

3. The switch with the lowest MAC address.

Switch Stack Configuration Files

The configuration files record these settings:

- System-level (global) configuration settings such as IP, STP, VLAN, and SNMP settings that apply to all stack members
- Stack member interface-specific configuration settings that are specific for each stack member



Note The interface-specific settings of the active switch are saved if the active switch is replaced without saving the running configuration to the startup configuration.

A new, out-of-box device joining a switch stack uses the system-level settings of that switch stack. If a device is moved to a different switch stack before it is powered on, that device loses its saved configuration file and

uses the system-level configuration of the new switch stack. If the device is powered on as a standalone device before it joins the new switch stack, the stack will reload. When the stack reloads, the new device may become the device, retain its configuration and overwrite the configuration files of the other stack members.

The interface-specific configuration of each stack member is associated with the stack member number. Stack members retain their numbers unless they are manually changed or they are already used by another member in the same switch stack. If the stack member number changes, the new number goes into effect after that stack member resets.

- If an interface-specific configuration does not exist for that member number, the stack member uses its default interface-specific configuration.
- If an interface-specific configuration exists for that member number, the stack member uses the interface-specific configuration associated with that member number.

If you replace a failed member with an identical model, the replacement member automatically uses the same interface-specific configuration as the failed device. You do not need to reconfigure the interface settings. The replacement device (referred to as the provisioned device) must have the same stack member number as the failed device.

You back up and restore the stack configuration in the same way as you would for a standalone device configuration.

Offline Configuration to Provision a Stack Member

You can use the offline configuration feature to *provision* (to supply a configuration to) a new switch before it joins the switch stack. You can configure the stack member number, the switch type, and the interfaces associated with a switch that is not currently part of the stack. The configuration that you create on the switch stack is called the *provisioned configuration*. The switch that is added to the switch stack and that receives this configuration is called the *provisioned switch*.

You manually create the provisioned configuration through the **switch stack-member-number provision type** global configuration command. You must change the *stack-member-number* on the provisioned switch before you add it to the stack, and it must match the stack member number that you created for the new switch on the switch stack. The switch type in the provisioned configuration must match the switch type of the newly added switch. The provisioned configuration is automatically created when a switch is added to a switch stack and when no provisioned configuration exists.

When you configure the interfaces associated with a provisioned switch, the switch stack accepts the configuration, and the information appears in the running configuration. However, as the switch is not active, any configuration on the interface is not operational and the interface associated with the provisioned switch does not appear in the display of the specific feature. For example, VLAN configuration information associated with a provisioned switch does not appear in the **show vlan** user EXEC command output on the switch stack.

The switch stack retains the provisioned configuration in the running configuration whether or not the provisioned switch is part of the stack. You can save the provisioned configuration to the startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. The startup configuration file ensures that the switch stack can reload and can use the saved information whether or not the provisioned switch is part of the switch stack.

Effects of Adding a Provisioned Switch to a Switch Stack

When you add a provisioned Device to the switch stack, the stack applies either the provisioned configuration or the default configuration. This table lists the events that occur when the switch stack compares the provisioned configuration with the provisioned switch.

Table 64: Results of Comparing the Provisioned Configuration with the Provisioned Switch

Scenario		Result
The stack member numbers and the Device types match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, and 2. If the Device type of the provisioned switch matches the Device type in the provisioned configuration on the stack. 	The switch stack applies the provisioned configuration to the provisioned switch and adds it to the stack.
The stack member numbers match but the Device types do not match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, but 2. The Device type of the provisioned switch does not match the Device type in the provisioned configuration on the stack. 	The switch stack applies the default configuration to the provisioned switch and adds it to the stack. The provisioned configuration is changed to reflect the new information.
The stack member number is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack. The provisioned configuration is changed to reflect the new information.
The stack member number of the provisioned switch is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack.

If you add a provisioned switch that is a different type than specified in the provisioned configuration to a powered-down switch stack and then apply power, the switch stack rejects the (now incorrect) **switch stack-member-number provision type** global configuration command in the startup configuration file. However, during stack initialization, the nondefault interface configuration information in the startup configuration file for the provisioned interfaces (potentially of the wrong type) is executed. Depending on the differences between the actual Device type and the previously provisioned switch type, some commands are rejected, and some commands are accepted.



Note If the switch stack does not contain a provisioned configuration for a new Device, the Device joins the stack with the default interface configuration. The switch stack then adds to its running configuration with a **switch stack-member-number provision type** global configuration command that matches the new Device. For configuration information, see the *Provisioning a New Member for a Switch Stack* section.

Effects of Replacing a Provisioned Switch in a Switch Stack

When a provisioned switch in a switch stack fails, it is removed from the stack, and is replaced with another Device, the stack applies either the provisioned configuration or the default configuration to it. The events that occur when the switch stack compares the provisioned configuration with the provisioned switch are the same as those when you add a provisioned switch to a stack.

Effects of Removing a Provisioned Switch from a Switch Stack

If you remove a provisioned switch from the switch stack, the configuration associated with the removed stack member remains in the running configuration as provisioned information. To completely remove the configuration, use the **no switch stack-member-number provision** global configuration command.

Stack Protocol Version

Each software image includes a *stack protocol version*. The stack protocol version has a *major* version number and a *minor* version number (for example 1.4, where 1 is the major version number and 4 is the minor version number). Both version numbers determine the level of compatibility among the stack members.

The switches with the same Cisco IOS software version have the same stack protocol version. Such switches are fully compatible, and all features function properly across the switch stack. A device with the same Cisco IOS software version as the active switch can immediately join the switch stack.

If an incompatibility exists, the fully functional stack members generate a system message that describes the cause of the incompatibility on the specific stack members. The active switch sends the message to all stack members.

For more information, see the *Major Version Number Incompatibility Among Switches* procedure and the *Minor Version Number Incompatibility Among Switches* procedure.

Major Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Device with different major Cisco IOS software versions usually have different stack protocol versions. Device with different major version numbers are incompatible and cannot exist in the same switch stack.

Minor Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Auto-Upgrade

The purpose of the auto-upgrade feature is to allow a switch to be upgraded to a compatible software image, so that the switch can join the switch stack.

When a new switch attempts to join a switch stack, each stack member performs compatibility checks with itself and the new switch. Each stack member sends the results of the compatibility checks to the active stack,

which uses the results to determine whether the switch can join the switch stack. If the software on the new switch is incompatible with the switch stack, the new switch enters version-mismatch (VM) mode.

If the auto-upgrade feature is enabled on the existing switch stack, the active stack automatically upgrades the new switch with the same software image running on a compatible stack member. Auto-upgrade starts a few minutes after the mismatched software is detected before starting.

Auto-upgrade includes an auto-copy process and an auto-extract process.

- Auto-copy automatically copies the software image running on any stack member to the new switch to automatically upgrade it. Auto-copy occurs if auto-upgrade is enabled, if there is enough flash memory in the new switch, and if the software image running on the switch stack is suitable for the new switch.



Note A switch in VM mode might not run all released software. For example, new switch hardware is not recognized in earlier versions of software.

When the auto-upgrade process is complete, the new switch reloads and joins the stack as a fully functioning member. If you have both stack cables connected during the reload, network downtime does not occur because the switch stack operates on two rings.

Switch Stack Management Connectivity

You manage the switch stack and the stack member interfaces through the active switch. You can use the CLI, SNMP, and any of the supported network management applications. You cannot manage stack members on an individual device basis.

Connectivity to Specific Stack Members

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

Connectivity to the Switch Stack Through an IP Address

The switch stack is managed through a single IP address. The IP address is a system-level setting and is not specific to the active stack or to any other stack member. You can still manage the stack through the same IP address even if you remove the active stack or any other stack member from the stack, provided there is IP connectivity.



Note Stack members retain their IP addresses when you remove them from a switch stack. To avoid a conflict by having two devices with the same IP address in your network, change the IP addresses of any active stack that you remove from the switch stack.

For related information about switch stack configurations, see the *Switch Stack Configuration Files* section.

Connectivity to the Switch Stack Through Console Ports

You can connect to the active switch by connecting a terminal or a PC to the active switch through the console port of one or more stack members.

When you use the console port of a stack member, a VTY session is created with the IP address in the 192.168.0.1/24 subnet.

Be careful when using multiple CLI sessions to the active switch. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.

We recommend using only one CLI session when managing the switch stack.

How to Configure a Switch Stack

Configuring a Network Port as Stack Port

You can configure both 10G network ports and multigig ports as stack ports or configure one port as stack port and retain another port as network port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **switch** *switch-number* **hstack-port** *stack-port*
4. **end**
5. **show switch horizontal-stack-ports**
6. **copy running-config startup-config**
7. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	switch <i>switch-number</i> hstack-port <i>stack-port</i> Example: <pre>Switch(config)# switch 1 hstack-port 1 TenGigabitEthernet 1/0/1</pre>	Configures the network port into a stack port . Note After configuration, restart the switch for network port to become a stack port.

	Command or Action	Purpose
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show switch horizontal-stack-ports Example: Switch# show switch hstack-ports	Verifies the operation status of network ports and stack ports.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.
Step 7	reload Example: Switch# reload	Reloads the configuration.

What to do next

To convert a stack port into a network port, run the **no switch *switch-number* hstack-port *stack-port*** command:

```
Switch(config)# no switch 1 hstack-port 1 TenGigabitEthernet 1/0/1
Switch# copy running-config startup-config
Switch# reload
```



Note After configuration, restart the switch to convert the stack port into a network port.



Note Stack port to network port conversion and vice versa cli is not nvgened. On write erase reload, the switch in stack mode will not be converted to standalone, and manual conversion of stack port to network port is required.

Enabling the Persistent MAC Address Feature



Note When you enter the command to configure this feature, a warning message appears with the consequences of your configuration. You should use this feature cautiously. Using the old active switch MAC address elsewhere in the same domain could result in lost traffic.

Follow these steps to enable persistent MAC address:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `stack-mac persistent timer [0 | time-value]`
4. `end`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>stack-mac persistent timer [0 <i>time-value</i>]</code></p> <p>Example:</p> <pre>Device(config)# stack-mac persistent timer 7</pre>	<p>Enables a time delay after a stack-active switch change before the stack MAC address changes to that of the new ac. If the previous active switch rejoins the stack during this period, the stack uses that MAC address as the stack MAC address.</p> <p>You can configure the time period as 0 to 60 minutes.</p> <ul style="list-style-type: none"> • Enter the command with no value to set the default delay of approximately 4 minutes. We recommend that you always enter a value. <p>If the command is entered without a value, the time delay appears in the running-config file with an explicit timer value of 4 minutes.</p> <ul style="list-style-type: none"> • Enter 0 to continue using the MAC address of the current active switch indefinitely.

	Command or Action	Purpose
		<p>The stack MAC address of the previous active switch is used until you enter the no stack-mac persistent timer command, which immediately changes the stack MAC address to that of the current active switch.</p> <ul style="list-style-type: none"> Enter a <i>time-value</i> from 1 to 60 minutes to configure the time period before the stack MAC address changes to the new active switch. <p>The stack MAC address of the previous active switch is used until the configured time period expires or until you enter the no stack-mac persistent timer command.</p> <p>Note If you enter the no stack-mac persistent timer command after a new active switch takes over, before the time expires, the switch stack moves to the current active switch MAC address.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no stack-mac persistent timer** global configuration command to disable the persistent MAC address feature.

Assigning a Stack Member Number

This optional task is available only from the active stack.

Follow these steps to assign a member number to a stack member:

SUMMARY STEPS

- enable**
- configure terminal**
- switch** *current-stack-member-number* **renumber** *new-stack-member-number*
- end**

5. **reload slot** *stack-member-number*
6. **show switch**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	switch <i>current-stack-member-number</i> renumber <i>new-stack-member-number</i> Example: <pre>Switch(config)# switch 3 renumber 4</pre>	Specifies the current stack member number and the new stack member number for the stack member. The range is 1 to 8. You can display the current stack member number by using the show switch user EXEC command.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	reload slot <i>stack-member-number</i> Example: <pre>Switch# reload slot 4</pre>	Resets the stack member.
Step 6	show switch Example: <pre>showSwitch</pre>	Verify the stack member number.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting the Stack Member Priority Value

This optional task is available only from the active stack.

Follow these steps to assign a priority value to a stack member:

SUMMARY STEPS

1. **enable**
2. **switch** *stack-member-number* **priority** *new-priority-number*
3. **show switch** *stack-member-number*
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	switch <i>stack-member-number</i> priority <i>new-priority-number</i> Example: Switch# switch 3 priority 2	Specifies the stack member number and the new priority for the stack member. The stack member number range is 1 to 8. The priority value range is 1 to 15. You can display the current priority value by using the show switch user EXEC command. The new priority value takes effect immediately but does not affect the current active stack. The new priority value helps determine which stack member is elected as the new active stack when the current active stack or switch stack resets.
Step 3	show switch <i>stack-member-number</i> Example: Switch# show switch	Verify the stack member priority value.
Step 4	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Provisioning a New Member for a Switch Stack

This optional task is available only from the active switch.

SUMMARY STEPS

1. **show switch**
2. **configure terminal**
3. **switch** *stack-member-number* **provision** *type*
4. **end**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	show switch Example: Switch# show switch	Displays summary information about the switch stack.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	switch <i>stack-member-number</i> provision <i>type</i> Example: Switch(config)# switch 3 provision WS-xxxx	Specifies the stack member number for the preconfigured switch. By default, no switches are provisioned. For <i>stack-member-number</i> , the range is 1 to 8. Specify a stack member number that is not already used in the switch stack. See Step 1. For <i>type</i> , enter the model number of a supported switch that is listed in the command-line help strings.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Removing Provisioned Switch Information

Before you begin, you must remove the provisioned switch from the stack. This optional task is available only from the active stack.

SUMMARY STEPS

1. **configure terminal**
2. **no switch** *stack-member-number* **provision**

3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	no switch <i>stack-member-number</i> provision Example: Switch(config)# no switch 3 provision	Removes the provisioning information for the specified member.
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 4	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Example

If you are removing a provisioned switch in a stack with this configuration:

- The stack has four members
- Stack member 1 is the active stack
- Stack member 3 is a provisioned switch

and want to remove the provisioned information and to avoid receiving an error message, you can remove power from stack member 3, disconnect the cables between the stack member 3 and switches to which it is connected, reconnect the cables between the remaining stack members, and enter the **no switch *stack-member-number* provision** global configuration command.

Troubleshooting the Switch Stack

Temporarily Disabling a Stack Port

If a stack port is flapping and causing instability in the stack ring, to disable the port, enter the **switch stack-member-number stack port port-number disable** privileged EXEC command. To reenables the port, enter the **switch stack-member-number stack port port-number enable** command.



Note Be careful when using the **switch stack-member-number stack port port-number disable** command. When you disable the stack port, the stack operates at half bandwidth.

A stack is in the full-ring state when all members are connected through the stack ports and are in the ready state.

The stack is in the partial-ring state when the following occurs:

- All members are connected through their stack ports but some are not in the ready state.
- Some members are not connected through the stack ports.

SUMMARY STEPS

1. **switch stack-member-number stack port port-number disable**
2. **switch stack-member-number stack port port-number enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch stack-member-number stack port port-number disable Example: Switch# switch 2 stack port 1 disable	Disables the specified stack port.
Step 2	switch stack-member-number stack port port-number enable Example: Switch# switch 2 stack port 1 enable	Reenables the stack port.

When you disable a stack port and the stack is in the full-ring state, you can disable only one stack port. This message appears:

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

When you disable a stack port and the stack is in the partial-ring state, you cannot disable the port. This message appears:

```
Disabling stack port not allowed with current stack configuration.
```

Reenabling a Stack Port While Another Member Starts

Stack Port 1 on Switch 1 is connected to Port 2 on Switch 4. If Port 1 is flapping, you can disable Port 1 with the **switch 1 stack port 1 disable** privileged EXEC command. While Port 1 on Switch 1 is disabled and Switch 1 is still powered on, follow these steps to reenoble a stack port:

-
- Step 1** Disconnect the stack cable between Port 1 on Switch 1 and Port 2 on Switch 4.
 - Step 2** Remove Switch 4 from the stack.
 - Step 3** Add a switch to replace Switch 4 and assign it switch-number 4.
 - Step 4** Reconnect the cable between Port 1 on Switch 1 and Port 2 on Switch 4 (the replacement switch).
 - Step 5** Reenable the link between the switches. Enter the **switch 1 stack port 1 enable** privileged EXEC command to enable Port 1 on Switch 1.
 - Step 6** Power on Switch 4.
-



Caution Powering on Switch 4 before enabling the Port 1 on Switch 1 might cause one of the switches to reload. If Switch 4 is powered on first, you might need to enter the **switch 1 stack port 1 enable** and the **switch 4 stack port 2 enable** privileged EXEC commands to bring up the link.

Monitoring the Device Stack

Table 65: Commands for Displaying Stack Information

Command	Description
show switch	Displays summary information about the stack, including the status of provisioned switches and switches in version-mismatch mode.
show switch <i>stack-member-number</i>	Displays information about a specific member.
show switch detail	Displays detailed information about the stack.
show switch neighbors	Displays the stack neighbors.
show switch stack-ports	Displays port information for the stack.

Configuration Examples for Switch Stacks

Switch Stack Configuration Scenarios

Most of these switch stack configuration scenarios assume that at least two device are connected through their ports.

Table 66: Configuration Scenarios

Scenario		Result
Active switch election specifically determined by existing active switches	Connect two powered-on switch stacks through the ports.	Only one of the two active switches becomes the new active switch.
Active switch election specifically determined by the stack member priority value	<ol style="list-style-type: none"> 1. Connect two switches through their ports. 2. Use the switch stack-member-number priority new-priority-number global configuration command to set one stack member with a higher member priority value. 3. Restart both stack members at the same time. 	The stack member with the higher priority value is elected active switch.
Active switch election specifically determined by the configuration file	<p>Assuming that both stack members have the same priority value:</p> <ol style="list-style-type: none"> 1. Make sure that one stack member has a default configuration and that the other stack member has a saved (nondefault) configuration file. 2. Restart both stack members at the same time. 	The stack member with the saved configuration file is elected active switch.
Active switch election specifically determined by the MAC address	Assuming that both stack members have the same priority value, configuration file, and feature set, restart both stack members at the same time.	The stack member with the lower MAC address is elected active switch.

Scenario		Result
Stack member number conflict	<p>Assuming that one stack member has a higher priority value than the other stack member:</p> <ol style="list-style-type: none"> 1. Ensure that both stack members have the same stack member number. If necessary, use the switch <i>current-stack-member-number</i> renumber <i>new-stack-member-number</i> global configuration command. 2. Restart both stack members at the same time. 	The stack member with the higher priority value retains its stack member number. The other stack member has a new stack member number.
Add a stack member	<ol style="list-style-type: none"> 1. Power off the new switch. 2. Through their ports, connect the new switch to a powered-on switch stack. 3. Power on the new switch. 	The active switch is retained. The new switch is added to the switch stack.
Active switch failure	Remove (or power off) the active switch.	One of the remaining stack members becomes the new active switch. All other stack members in the stack remain as stack members and do not reboot.
Add more than eight stack members	<ol style="list-style-type: none"> 1. Through their ports, connect nine device. 2. Power on all device. 	<p>Two device become active switches. One active switch has eight stack members. The other active switch remains as a standalone device.</p> <p>Use the Mode button and port LEDs on the device to identify which device are active switches and which device belong to each active switch.</p>

Enabling the Persistent MAC Address Feature: Example

This example shows how to configure the persistent MAC address feature for a 7-minute time delay and to verify the configuration:

```
Switch(config)# stack-mac persistent timer 7
WARNING: The stack continues to use the base MAC of the old Master
WARNING: as the stack MAC after a master switchover until the MAC
WARNING: persistency timer expires. During this time the Network
WARNING: Administrators must make sure that the old stack-mac does
WARNING: not appear elsewhere in this network domain. If it does,
WARNING: user traffic may be blackholed.
Switch(config)# end
Switch# show switch
Switch/Stack Mac Address : 0016.4727.a900
```



```

Mac persistency wait time: 7 mins
Switch# show running-config | section hstack-ports
-----
Switch# Role Mac Address Priority Version H/W Current State
-----
*1 Master 0016.4727.a900 1 P2B Ready

```

Provisioning a New Member for a Switch Stack: Example

This example shows how to provision a switch with a stack member number of 2 for the switch stack. The **show running-config** command output shows the interfaces associated with the provisioned switch:

Configuring a Network Port into a Stack Port: Examples

The following example shows how to convert a network port to stack port.

```

Switch> enable
Switch#configure terminal
Switch(config)#switch 1 hstack-port 1 TenGigabitEthernet 1/0/1
Do you want to continue?[confirm]
New port setting will be effective after next reload

```

```

Switch(config)#switch 1 hstack-port 2 TenGigabitEthernet 1/0/2
Do you want to continue?[confirm]
New port setting will be effective after next reload

```

The following output example shows the status of the port before reload from network port to stack port.

```

Switch#show switch hstack-ports
Horizontal stack port status :
Te Ports Stack Port Operational Status Next Reload Status Media Type
-----
Tel/0/1 1 N/W Port Stack Port Fiber
Tel/0/2 2 N/W Port Stack Port Fiber

```

The following output example shows the status of the port after reload from network port to stack port.

```

Switch#show switch hstack-ports
Horizontal stack port status :
Te Ports Stack Port Operational Status Next Reload Status Media Type
-----
Tel/0/1 1 Stack Port Stack Port Fiber
Tel/0/2 2 Stack Port Stack Port Fiber

```

The following example shows how to convert a stack port back to network port.

```

Switch> enable
Switch#configure terminal
Switch(config)#no switch 1 hstack-port 1
Do you want to continue?[confirm]
New port setting will be effective after next reload

```

The following output example shows the status of the port before reload from stack port to network port.

```

Switch#show switch hstack-ports
Horizontal stack port status :
Te Ports Stack Port Operational Status Next Reload Status Media Type
-----

```

```

Te1/0/1      1      Stack Port      N/W Port      Fiber
Te1/0/2      2      Stack Port      Stack Port     Fiber

```

The following output example shows the status of the port after reload from stack port to network port.

```

Switch#show switch hstack-ports
Horizontal stack port status :
Te Ports   Stack Port   Operational Status   Next Reload Status   Media Type
-----
Te1/0/1    1            N/W Port             N/W Port             Fiber
Te1/0/2    2            Stack Port           Stack Port           Fiber

```

The following output example shows the status of the horizontal stack port.

```

Switch# show switch hstack-ports
Horizontal stack port status :
Te Ports   Stack Port   Operational Status   Next Reload Status   Media Type
-----
Te1/0/1    1            Stack Port          Stack Port           Fiber
Te1/0/2    2            Stack Port          Stack Port           Fiber
Te2/0/1    1            Stack Port          Stack Port           Fiber
Te2/0/2    2            Stack Port          Stack Port           Fiber
Te3/0/1    1            Stack Port          Stack Port           Copper
Te3/0/2    NA           N/W Port           N/W Port            Copper
Te3/0/3    2            Stack Port          Stack Port           Fiber
Te3/0/4    NA           N/W Port           N/W Port            Fiber
Te4/0/1    NA           N/W Port           N/W Port            Copper
Te4/0/2    1            Stack Port          Stack Port           Copper
Te4/0/3    2            Stack Port          Stack Port           Fiber
Te4/0/4    NA           N/W Port           N/W Port            Fiber
Te5/0/1    1            Stack Port          Stack Port           Fiber
Te5/0/2    2            Stack Port          Stack Port           Fiber
Te6/0/1    1            Stack Port          Stack Port           Fiber
Te6/0/2    2            Stack Port          Stack Port           Fiber
Te7/0/1    1            Stack Port          Stack Port           Copper
Te7/0/2    NA           N/W Port           N/W Port            Copper
Te7/0/3    2            Stack Port          Stack Port           Fiber
Te7/0/4    NA           N/W Port           N/W Port            Fiber
Te8/0/1    NA           N/W Port           N/W Port            Copper
Te8/0/2    1            Stack Port          Stack Port           Copper
Te8/0/3    2            Stack Port          N/W Port            Fiber
Te8/0/4    NA           N/W Port           N/W Port            Fiber

```

Additional References for Switch Stacks

Related Documents

Related Topic	Document Title
Cabling and powering on a switch stack.	

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and , use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



PART VI

Network Management

- [Configuring Cisco IOS Configuration Engine, on page 703](#)
- [Configuring the Cisco Discovery Protocol, on page 721](#)
- [Configuring Simple Network Management Protocol, on page 731](#)
- [Configuring SPAN and RSPAN, on page 755](#)
- [Configuring RMON, on page 793](#)
- [Configuring Embedded Event Manager, on page 801](#)
- [Configuring Flexible NetFlow, on page 809](#)
- [Configuring Cache Services Using the Web Cache Communication Protocol, on page 833](#)



CHAPTER 37

Configuring Cisco IOS Configuration Engine

- [Prerequisites for Configuring the Configuration Engine, on page 703](#)
- [Restrictions for Configuring the Configuration Engine, on page 703](#)
- [Information About Configuring the Configuration Engine, on page 703](#)
- [How to Configure the Configuration Engine, on page 709](#)
- [Monitoring CNS Configurations, on page 720](#)

Prerequisites for Configuring the Configuration Engine

- Obtain the name of the configuration engine instance to which you are connecting.
- Because the CNS uses both the event bus and the configuration server to provide configurations to devices, you must define both ConfigID and Device ID for each configured device.
- All devices configured with the **cns config partial** global configuration command must access the event bus. The DeviceID, as originated on the device, must match the DeviceID of the corresponding device definition in the Cisco Configuration Engine. You must know the hostname of the event bus to which you are connecting.

Restrictions for Configuring the Configuration Engine

- Within the scope of a single instance of the configuration server, no two configured devices can share the same value for ConfigID.
- Within the scope of a single instance of the event bus, no two configured devices can share the same value for DeviceID.

Information About Configuring the Configuration Engine

Cisco Configuration Engine Software

The Cisco Configuration Engine is network management utility software that acts as a configuration service for automating the deployment and management of network devices and services. Each Cisco Configuration

Engine manages a group of Cisco devices (devices and routers) and the services that they deliver, storing their configurations and delivering them as needed. The Cisco Configuration Engine automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

The Cisco Configuration Engine supports standalone and server modes and has these Cisco Networking Services (CNS) components:

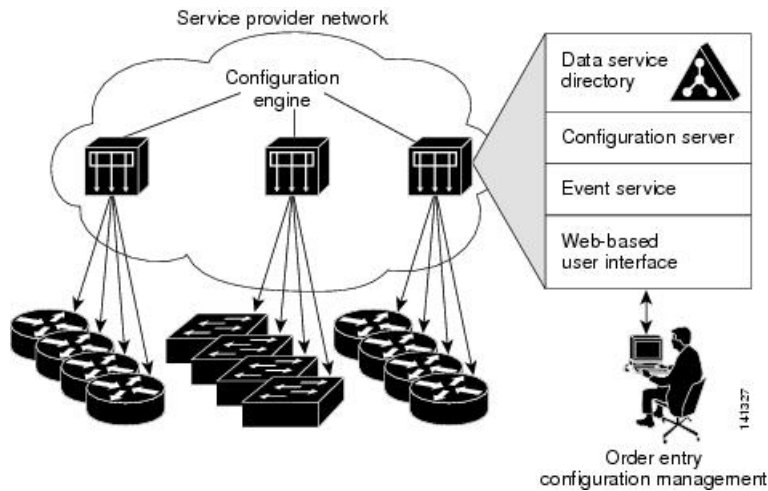
- Configuration service:
 - Web server
 - File manager
 - Namespace mapping server
- Event service (event gateway)
- Data service directory (data models and schema)



Note Support for Cisco Configuration Engine will be deprecated in future releases. Use the configuration described in [Cisco Plug and Play Feature Guide](#).

In standalone mode, the Cisco Configuration Engine supports an embedded directory service. In this mode, no external directory or other data store is required. In server mode, the Cisco Configuration Engine supports the use of a user-defined external directory.

Figure 63: Cisco Configuration Engine Architectural Overview



Configuration Service

The Configuration Service is the core component of the Cisco Configuration Engine. It consists of a Configuration Server that works with Cisco IOS CNS agents on the device. The Configuration Service delivers device and service configurations to the device for initial configuration and mass reconfiguration by logical

groups. Devices receive their initial configuration from the Configuration Service when they start up on the network for the first time.

The Configuration Service uses the CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The Configuration Server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified by using Lightweight Directory Access Protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The Cisco IOS agent can perform a syntax check on received configuration files and publish events to show the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

Event Service

The Cisco Configuration Engine uses the Event Service for receipt and generation of configuration events. The Event Service consists of an event agent and an event gateway. The event agent is on the device and facilitates the communication between the device and the event gateway on the Cisco Configuration Engine.

The Event Service is a highly capable publish-and-subscribe communication method. The Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

NameSpace Mapper

The Cisco Configuration Engine includes the NameSpace Mapper (NSM) that provides a lookup service for managing logical groups of devices based on application, device or group ID, and event.

Cisco IOS devices recognize only event subject-names that match those configured in Cisco IOS software; for example, `cisco.cns.config.load`. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM changes your event subject-name strings to those known by Cisco IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

Cisco Networking Services IDs and Device Hostnames

The Cisco Configuration Engine assumes that a unique identifier is associated with each configured device. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

The Cisco Configuration Engine intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term *ConfigID* is the unique identifier for a device. Within the scope of the event bus namespace, the term *DeviceID* is the CNS unique identifier for a device.

ConfigID

Each configured device has a unique ConfigID, which serves as the key into the Cisco Configuration Engine directory for the corresponding set of device CLI attributes. The ConfigID defined on the device must match the ConfigID for the corresponding device definition on the Cisco Configuration Engine.

The ConfigID is fixed at startup time and cannot be changed until the device restarts, even if the device hostname is reconfigured.

DeviceID

Each configured device participating on the event bus has a unique DeviceID, which is analogous to the device source address so that the device can be targeted as a specific destination on the bus.

The origin of the DeviceID is defined by the Cisco IOS hostname of the device. However, the DeviceID variable and its usage reside within the event gateway adjacent to the device.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the device. The event gateway represents the device and its corresponding DeviceID to the event bus.

The device declares its hostname to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the DeviceID value to the Cisco IOS hostname each time this connection is established. The event gateway retains this DeviceID value for the duration of its connection to the device.

Hostname and DeviceID

The DeviceID is fixed at the time of the connection to the event gateway and does not change even when the device hostname is reconfigured.

When changing the device hostname on the device, the only way to refresh the DeviceID is to break the connection between the device and the event gateway. For instructions on refreshing DeviceIDs, see "Related Topics."

When the connection is reestablished, the device sends its modified hostname to the event gateway. The event gateway redefines the DeviceID to the new value.



Caution When using the Cisco Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the device acquires *after*, not *before*, and you must reinitialize the configuration for your Cisco IOS CNS agent. Otherwise, subsequent partial configuration command operations may malfunction.

Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a device, the configuration server uses the hostname as the DeviceID when an event is sent on hostname. If the hostname has not been set, the event is sent on the `cn=<value>` of the device.

In server mode, the hostname is not used. In this mode, the unique DeviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the device.

These and other associated attributes (tag value pairs) are set when you run **Setup** on the Cisco Configuration Engine.

Cisco IOS CNS Agents

The CNS event agent feature allows the device to publish and subscribe to events on the event bus and works with the Cisco IOS CNS agent. These agents, embedded in the device Cisco IOS software, allow the device to be connected and automatically configured.

Initial Configuration

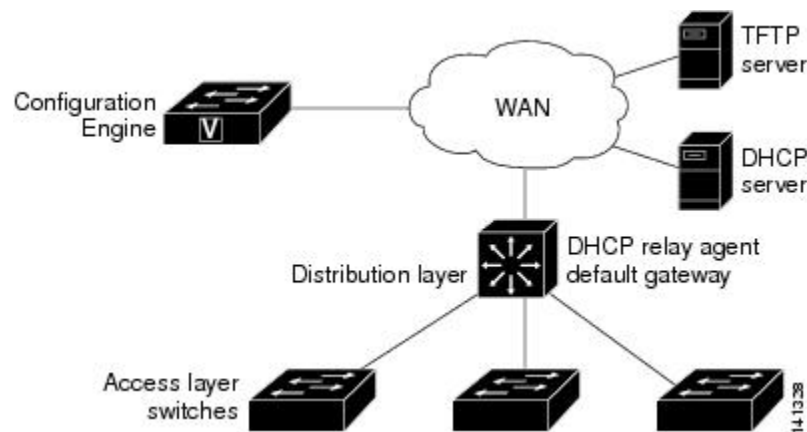
When the device first comes up, it attempts to get an IP address by broadcasting a Dynamic Host Configuration Protocol (DHCP) request on the network. Assuming there is no DHCP server on the subnet, the distribution device acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new device and includes the Trivial File Transfer Protocol (TFTP) server Internet Protocol (IP) address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the device.

The device automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the device loads the file in its running configuration.

The Cisco IOS CNS agents initiate communication with the Configuration Engine by using the appropriate ConfigID and EventID. The Configuration Engine maps the Config ID to a template and downloads the full configuration file to the device.

The following figure shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 64: Initial Configuration



Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS CNS agent. Incremental (partial) configurations can be sent to the device. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the device to initiate a pull operation.

The device can check the syntax of the configuration before applying it. If the syntax is correct, the device applies the incremental configuration and publishes an event that signals success to the configuration server. If the device does not apply the incremental configuration, it publishes an event showing an error status. When the device has applied the incremental configuration, it can write it to nonvolatile random-access memory (NVRAM) or wait until signaled to do so.

Synchronized Configuration

When the device receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the device not to save the updated configuration into its NVRAM. The device uses the updated configuration as its running configuration. This ensures that the device configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

Automated CNS Configuration

To enable automated CNS configuration of the device, you must first complete the prerequisites listed in this topic. When you complete them, power on the device. At the **setup** prompt, do nothing; the device begins the initial configuration. When the full configuration file is loaded on your device, you do not need to do anything else.

For more information on what happens during initial configuration, see "Related Topics."

Table 67: Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access device	Factory default (no configuration file)
Distribution device	<ul style="list-style-type: none"> • IP helper address • Enable DHCP relay agent² • IP routing (if used as default gateway)
DHCP server	<ul style="list-style-type: none"> • IP address assignment • TFTP server IP address • Path to bootstrap configuration file on the TFTP server • Default gateway IP address
TFTP server	<ul style="list-style-type: none"> • A bootstrap configuration file that includes the CNS configuration commands that enable the device to communicate with the Configuration Engine • The device configured to use either the device MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID • The CNS event agent configured to push the configuration file to the device
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template.

² A DHCP Relay is needed only when the DHCP Server is on a different subnet from the client.

How to Configure the Configuration Engine

Enabling the CNS Event Agent



Note You must enable the CNS event agent on the device before you enable the CNS configuration agent.

Follow these steps to enable the CNS event agent on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns event** *{hostname | ip-address}* [*port-number*] [**keepalive** *seconds* *retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	cns event <i>{hostname ip-address}</i> [<i>port-number</i>] [keepalive <i>seconds</i> <i>retry-count</i>] [failover-time <i>seconds</i>] [reconnect-time <i>time</i>] backup Example: <pre>Switch(config)# cns event 10.180.1.27 keepalive 120 10</pre>	Enables the event agent, and enters the gateway parameters. <ul style="list-style-type: none"> • For <i>{hostname ip-address}</i>, enter either the hostname or the IP address of the event gateway. • (Optional) For <i>port number</i>, enter the port number for the event gateway. The default port number is 11011. • (Optional) For keepalive <i>seconds</i>, enter how often the device sends keepalive messages. For <i>retry-count</i>, enter the number of unanswered keepalive messages that the device sends before the connection is terminated. The default for each is 0.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) For failover-time <i>seconds</i>, enter how long the device waits for the primary gateway route after the route to the backup gateway is established. • (Optional) For reconnect-time <i>time</i>, enter the maximum time interval that the device waits before trying to reconnect to the event gateway. • (Optional) Enter backup to show that this is the backup gateway. (If omitted, this is the primary gateway.) <p>Note Though visible in the command-line help string, the encrypt and the clock-timeout <i>time</i> keywords are not supported.</p>
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the event agent, use the **show cns event connections** command in privileged EXEC mode.

To disable the CNS event agent, use the **no cns event** { *ip-address* | *hostname* } global configuration command.

Enabling the Cisco IOS CNS Agent

Follow these steps to enable the Cisco IOS CNS agent on the device.

Before you begin

You must enable the CNS event agent on the device before you enable this agent.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns config initial** *{hostname | ip-address}* [*port-number*]
4. **cns config partial** *{hostname | ip-address}* [*port-number*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**
8. Start the Cisco IOS CNS agent on the device.

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	cns config initial <i>{hostname ip-address}</i> [<i>port-number</i>] Example: <pre>Switch(config)# cns config initial 10.180.1.27 10</pre>	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. <ul style="list-style-type: none"> • For <i>{hostname ip-address}</i>, enter either the hostname or the IP address of the configuration server. • (Optional) For <i>port number</i>, enter the port number for the configuration server. This command enables the Cisco IOS CNS agent and initiates an initial configuration on the device.
Step 4	cns config partial <i>{hostname ip-address}</i> [<i>port-number</i>] Example: <pre>Switch(config)# cns config partial 10.180.1.27 10</pre>	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. <ul style="list-style-type: none"> • For <i>{hostname ip-address}</i>, enter either the hostname or the IP address of the configuration server. • (Optional) For <i>port number</i>, enter the port number for the configuration server. Enables the Cisco IOS CNS agent and initiates a partial configuration on the device.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.
Step 8	Start the Cisco IOS CNS agent on the device.	

What to do next

You can now use the Cisco Configuration Engine to remotely send incremental configurations to the device.

Enabling an Initial Configuration for Cisco IOS CNS Agent

Follow these steps to enable the CNS configuration agent and initiate an initial configuration on the device.

SUMMARY STEPS

- enable**
- configure terminal**
- cns template connect** *name*
- cli** *config-text*
- Repeat Steps 3 to 4 to configure another CNS connect template.
- exit**
- cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]
- discover** {**controller** *controller-type* | **dcli** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}
- template** *name* [... *name*]
- Repeat Steps 8 to 9 to specify more interface parameters and CNS connect templates in the CNS connect profile.
- exit**
- hostname** *name*
- ip route** *network-number*
- cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]
- cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]
- cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]
- end**

18. `show running-config`
19. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>cns template connect <i>name</i></p> <p>Example:</p> <pre>Switch(config)# cns template connect template-dhcp</pre>	<p>Enters CNS template connect configuration mode, and specifies the name of the CNS connect template.</p>
Step 4	<p>cli <i>config-text</i></p> <p>Example:</p> <pre>Switch(config-tmpl-conn)# cli ip address dhcp</pre>	<p>Enters a command line for the CNS connect template. Repeat this step for each command line in the template.</p>
Step 5	<p>Repeat Steps 3 to 4 to configure another CNS connect template.</p>	
Step 6	<p>exit</p> <p>Example:</p> <pre>Switch(config)# exit</pre>	<p>Returns to global configuration mode.</p>
Step 7	<p>cns connect <i>name</i> [<i>retries number</i>] [<i>retry-interval seconds</i>] [<i>sleep seconds</i>] [<i>timeout seconds</i>]</p> <p>Example:</p> <pre>Switch(config)# cns connect dhcp</pre>	<p>Enters CNS connect configuration mode, specifies the name of the CNS connect profile, and defines the profile parameters. The device uses the CNS connect profile to connect to the Configuration Engine.</p> <ul style="list-style-type: none"> • Enter the <i>name</i> of the CNS connect profile. • (Optional) For retries number, enter the number of connection retries. The range is 1 to 30. The default is 3. • (Optional) For retry-interval seconds, enter the interval between successive connection attempts to the Configuration Engine. The range is 1 to 40 seconds. The default is 10 seconds.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) For sleep <i>seconds</i>, enter the amount of time before which the first connection attempt occurs. The range is 0 to 250 seconds. The default is 0. • (Optional) For timeout <i>seconds</i>, enter the amount of time after which the connection attempts end. The range is 10 to 2000 seconds. The default is 120.
Step 8	<p>discover {controller <i>controller-type</i> dcli [subinterface <i>subinterface-number</i>] interface [<i>interface-type</i>] line <i>line-type</i>}</p> <p>Example:</p> <pre>Switch(config-cns-conn)# discover interface gigabitethernet</pre>	<p>Specifies the interface parameters in the CNS connect profile.</p> <ul style="list-style-type: none"> • For controller <i>controller-type</i>, enter the controller type. • For dcli, enter the active data-link connection identifiers (DLCIs). <p>(Optional) For subinterface <i>subinterface-number</i>, specify the point-to-point subinterface number that is used to search for active DLCIs.</p> <ul style="list-style-type: none"> • For interface [<i>interface-type</i>], enter the type of interface. • For line <i>line-type</i>, enter the line type.
Step 9	<p>template <i>name</i> [... <i>name</i>]</p> <p>Example:</p> <pre>Switch(config-cns-conn)# template template-dhcp</pre>	<p>Specifies the list of CNS connect templates in the CNS connect profile to be applied to the device configuration. You can specify more than one template.</p>
Step 10	Repeat Steps 8 to 9 to specify more interface parameters and CNS connect templates in the CNS connect profile.	
Step 11	<p>exit</p> <p>Example:</p> <pre>Switch(config-cns-conn)# exit</pre>	Returns to global configuration mode.
Step 12	<p>hostname <i>name</i></p> <p>Example:</p> <pre>Switch(config)# hostname device1</pre>	Enters the hostname for the device.
Step 13	<p>ip route <i>network-number</i></p> <p>Example:</p> <pre>RemoteSwitch(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1</pre>	(Optional) Establishes a static route to the Configuration Engine whose IP address is <i>network-number</i> .

	Command or Action	Purpose
Step 14	<p>cns id <i>interface num</i> {dns-reverse ipaddress mac-address} [event] [image]</p> <p>Example:</p> <pre>RemoteSwitch(config)# cns id GigabitEthernet0/1 ipaddress</pre>	<p>(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the cns id {hardware-serial hostname string string udi} [event] [image] command.</p> <ul style="list-style-type: none"> For <i>interface num</i>, enter the type of interface. For example, ethernet, group-async, loopback, or virtual-template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID. For {dns-reverse ipaddress mac-address}, enter dns-reverse to retrieve the hostname and assign it as the unique ID, enter ipaddress to use the IP address, or enter mac-address to use the MAC address as the unique ID. (Optional) Enter event to set the ID to be the event-id value used to identify the device. (Optional) Enter image to set the ID to be the image-id value used to identify the device. <p>Note If both the event and image keywords are omitted, the image-id value is used to identify the device.</p>
Step 15	<p>cns id {hardware-serial hostname string string udi} [event] [image]</p> <p>Example:</p> <pre>RemoteSwitch(config)# cns id hostname</pre>	<p>(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the cns id <i>interface num</i> {dns-reverse ipaddress mac-address} [event] [image] command.</p> <ul style="list-style-type: none"> For { hardware-serial hostname string string udi }, enter hardware-serial to set the device serial number as the unique ID, enter hostname (the default) to select the device hostname as the unique ID, enter an arbitrary text string for string string as the unique ID, or enter udi to set the unique device identifier (UDI) as the unique ID.
Step 16	<p>cns config initial {<i>hostname</i> <i>ip-address</i>} [<i>port-number</i>] [event] [no-persist] [page page] [source ip-address] [syntax-check]</p> <p>Example:</p> <pre>RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist</pre>	<p>Enables the Cisco IOS agent, and initiates an initial configuration.</p> <ul style="list-style-type: none"> For {<i>hostname</i> <i>ip-address</i>}, enter the hostname or the IP address of the configuration server. (Optional) For <i>port-number</i>, enter the port number of the configuration server. The default port number is 80.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) Enable event for configuration success, failure, or warning messages when the configuration is finished. • (Optional) Enable no-persist to suppress the automatic writing to NVRAM of the configuration pulled as a result of entering the cns config initial global configuration command. If the no-persist keyword is not entered, using the cns config initial command causes the resultant configuration to be automatically written to NVRAM. • (Optional) For page page, enter the web page of the initial configuration. The default is <code>/Config/config/asp</code>. • (Optional) Enter source ip-address to use for source IP address. • (Optional) Enable syntax-check to check the syntax when this parameter is entered. <p>Note Though visible in the command-line help string, the encrypt, status url, and inventory keywords are not supported.</p>
Step 17	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 18	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 19	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the configuration agent, use the **show cns config connections** command in privileged EXEC mode.

To disable the CNS Cisco IOS agent, use the **no cns config initial { ip-address | hostname }** global configuration command.

Refreshing DeviceIDs

Follow these steps to refresh a DeviceID when changing the hostname on the device.

SUMMARY STEPS

1. **enable**
2. **show cns config connections**
3. Make sure that the CNS event agent is properly connected to the event gateway.
4. **show cns event connections**
5. Record from the output of Step 4 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.
6. **configure terminal**
7. **no cns event ip-address port-number**
8. **cns event ip-address port-number**
9. **end**
10. Make sure that you have reestablished the connection between the device and the event connection by examining the output from **show cns event connections**.
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show cns config connections Example: Switch# show cns config connections	Displays whether the CNS event agent is connecting to the gateway, connected, or active, and the gateway used by the event agent, its IP address and port number.
Step 3	Make sure that the CNS event agent is properly connected to the event gateway.	Examine the output of show cns config connections for the following: <ul style="list-style-type: none"> • Connection is active. • Connection is using the currently configured device hostname. The DeviceID will be refreshed to correspond to the new hostname configuration using these instructions.
Step 4	show cns event connections Example: Switch# show cns event connections	Displays the event connection information for your device.

	Command or Action	Purpose
Step 5	Record from the output of Step 4 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.	
Step 6	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 7	no cns event ip-address port-number Example: Switch(config)# <code>no cns event 172.28.129.22 2012</code>	Specifies the IP address and port number that you recorded in Step 5 in this command. This command breaks the connection between the device and the event gateway. It is necessary to first break, then reestablish, this connection to refresh the DeviceID.
Step 8	cns event ip-address port-number Example: Switch(config)# <code>cns event 172.28.129.22 2012</code>	Specifies the IP address and port number that you recorded in Step 5 in this command. This command reestablishes the connection between the device and the event gateway.
Step 9	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 10	Make sure that you have reestablished the connection between the device and the event connection by examining the output from show cns event connections .	
Step 11	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 12	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling a Partial Configuration for Cisco IOS CNS Agent

Follow these steps to enable the Cisco IOS CNS agent and to initiate a partial configuration on the device.

SUMMARY STEPS

1. enable
2. configure terminal
3. cns config partial {ip-address | hostname} [port-number] [source ip-address]
4. end
5. show running-config
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>cns config partial {ip-address hostname} [port-number] [source ip-address]</p> <p>Example:</p> <pre>Switch(config)# cns config partial 172.28.129.22 2013</pre>	<p>Enables the configuration agent, and initiates a partial configuration.</p> <ul style="list-style-type: none"> • For {ip-address hostname}, enter the IP address or the hostname of the configuration server. • (Optional) For port-number, enter the port number of the configuration server. The default port number is 80. • (Optional) Enter source ip-address to use for the source IP address. <p>Note Though visible in the command-line help string, the encrypt keyword is not supported.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the configuration agent, use either the **show cns config stats** or the **show cns config outstanding** command in privileged EXEC mode.

To disable the Cisco IOS agent, use the **no cns config partial** { *ip-address* | *hostname* } global configuration command. To cancel a partial configuration, use the **cns config cancel** global configuration command.

Monitoring CNS Configurations

Table 68: CNS show Commands

Command	Purpose
show cns config connections Switch# <code>show cns config connections</code>	Displays the status of the CNS Cisco IOS CNS agent connections.
show cns config outstanding Switch# <code>show cns config outstanding</code>	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
show cns config stats Switch# <code>show cns config stats</code>	Displays statistics about the Cisco IOS CNS agent.
show cns event connections Switch# <code>show cns event connections</code>	Displays the status of the CNS event agent connections.
show cns event gateway Switch# <code>show cns event gateway</code>	Displays the event gateway information for your device.
show cns event stats Switch# <code>show cns event stats</code>	Displays statistics about the CNS event agent.
show cns event subject Switch# <code>show cns event subject</code>	Displays a list of event agent subjects that are subscribed to by applications.



CHAPTER 38

Configuring the Cisco Discovery Protocol

Cisco Discovery Protocol is a Layer 2, media-independent, and network-independent protocol that runs on Cisco devices and enables networking applications to learn about directly connected devices nearby. This protocol facilitates the management of Cisco devices by discovering these devices, determining how they are configured, and allowing systems using different network-layer protocols to learn about each other.

This module describes Cisco Discovery Protocol Version 2 and how it functions with SNMP.

- [Information About CDP, on page 721](#)
- [How to Configure CDP, on page 722](#)
- [Monitoring and Maintaining Cisco Discovery Protocol, on page 729](#)

Information About CDP

Cisco Discovery Protocol Overview

Cisco Discovery Protocol is a device discovery protocol that runs over Layer 2 (the data-link layer) on all Cisco-manufactured devices (routers, bridges, access servers, controllers, and switches) and allows network management applications to discover Cisco devices that are neighbors of already known devices. With Cisco Discovery Protocol, network management applications can learn the device type and the SNMP agent address of neighboring devices running lower-layer, transparent protocols. This feature enables applications to send SNMP queries to neighboring devices.

Cisco Discovery Protocol runs on all media that support Subnetwork Access Protocol (SNAP). Because Cisco Discovery Protocol runs over the data-link layer only, two systems that support different network-layer protocols can learn about each other.

Each Cisco Discovery Protocol-configured device sends periodic messages to a multicast address, advertising at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime information, which is the length of time a receiving device holds Cisco Discovery Protocol information before discarding it. Each device also listens to the messages sent by other devices to learn about neighboring devices.

On the device, Cisco Discovery Protocol enables Network Assistant to display a graphical view of the network. The device uses Cisco Discovery Protocol to find cluster candidates and maintain information about cluster members and other devices up to three cluster-enabled devices away from the command device by default.

- Cisco Discovery Protocol identifies connected endpoints that communicate directly with the device.

- To prevent duplicate reports of neighboring devices, only one wired device reports the location information.
- The wired device and the endpoints both send and receive location information.

Default Cisco Discovery Protocol Configuration

This table shows the default Cisco Discovery Protocol configuration.

Feature	Default Setting
Cisco Discovery Protocol global state	Enabled
Cisco Discovery Protocol interface state	Enabled
Cisco Discovery Protocol timer (packet update frequency)	60 seconds
Cisco Discovery Protocol holdtime (before discarding)	180 seconds
Cisco Discovery Protocol Version-2 advertisements	Enabled

How to Configure CDP

Configuring Cisco Discovery Protocol Characteristics

You can configure these Cisco Discovery Protocol characteristics:

- Frequency of Cisco Discovery Protocol updates
- Amount of time to hold the information before discarding it
- Whether or not to send Version 2 advertisements



Note Steps 3 through 5 are all optional and can be performed in any order.

Follow these steps to configure the Cisco Discovery Protocol characteristics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cdp timer** *seconds*
4. **cdp holdtime** *seconds*
5. **cdp advertise-v2**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	cdp timer <i>seconds</i> Example: Switch(config)# cdp timer 20	(Optional) Sets the transmission frequency of Cisco Discovery Protocol updates in seconds. The range is 5 to 254; the default is 60 seconds.
Step 4	cdp holdtime <i>seconds</i> Example: Switch(config)# cdp holdtime 60	(Optional) Specifies the amount of time a receiving device should hold the information sent by your device before discarding it. The range is 10 to 255 seconds; the default is 180 seconds.
Step 5	cdp advertise-v2 Example: Switch(config)# cdp advertise-v2	(Optional) Configures Cisco Discovery Protocol to send Version 2 advertisements. This is the default state.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Switch# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no** form of the Cisco Discovery Protocol commands to return to the default settings.

Disabling Cisco Discovery Protocol

Cisco Discovery Protocol is enabled by default.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.

Follow these steps to disable the Cisco Discovery Protocol device discovery capability.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	no cdp run Example: Switch(config)# no cdp run	Disables Cisco Discovery Protocol.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

You must reenable Cisco Discovery Protocol to use it.

Enabling Cisco Discovery Protocol

Cisco Discovery Protocol is enabled by default.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.

Follow these steps to enable Cisco Discovery Protocol when it has been disabled.

Before you begin

Cisco Discovery Protocol must be disabled, or it cannot be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	cdp run Example: Switch(config)# cdp run	Enables Cisco Discovery Protocol if it has been disabled.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **show run all** command to show that Cisco Discovery Protocol has been enabled. If you enter only **show run**, the enabling of Cisco Discovery Protocol may not be displayed.

Disabling Cisco Discovery Protocol on an Interface

Cisco Discovery Protocol is enabled by default on all supported interfaces to send and to receive Cisco Discovery Protocol information.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.



Note Cisco Discovery Protocol bypass is not supported and may cause a port go into err-disabled state.

Follow these steps to disable Cisco Discovery Protocol on a port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no cdp enable**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies the interface on which you are disabling Cisco Discovery Protocol, and enters interface configuration mode.
Step 4	no cdp enable Example: Switch(config-if)# no cdp enable	Disables Cisco Discovery Protocol on the interface specified in Step 3.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show running-config</code>	
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling Cisco Discovery Protocol on an Interface

Cisco Discovery Protocol is enabled by default on all supported interfaces to send and to receive Cisco Discovery Protocol information.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.



Note Cisco Discovery Protocol bypass is not supported and may cause a port go into err-disabled state.

Follow these steps to enable Cisco Discovery Protocol on a port on which it has been disabled.

Before you begin

Cisco Discovery Protocol must be disabled on the port that you are trying to Cisco Discovery Protocol enable on, or it cannot be enabled.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `cdp enable`
5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	<code>Switch> enable</code>	
Step 2	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <code>Switch(config)# interface gigabitethernet1/0/1</code>	Specifies the interface on which you are enabling Cisco Discovery Protocol, and enters interface configuration mode.
Step 4	cdp enable Example: <code>Switch(config-if)# cdp enable</code>	Enables Cisco Discovery Protocol on a disabled interface.
Step 5	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <code>Switch# show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining Cisco Discovery Protocol

Table 69: Commands for Displaying Cisco Discovery Protocol Information

Command	Description
<code>clear cdp counters</code>	Resets the traffic counters to zero.
<code>clear cdp table</code>	Deletes the Cisco Discovery Protocol table of information about
<code>show cdp</code>	Displays global information, such as frequency of transmissions and for packets being sent.

Command	Description
show cdp entry <i>entry-name</i> [version] [protocol]	Displays information about a specific neighbor. You can enter an asterisk (*) to display all Cisco Discovery Protocol neighbors or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [<i>interface-id</i>]	Displays information about interfaces where Cisco Discovery Protocol is enabled. You can limit the display to the interface about which you want information.
show cdp neighbors [<i>interface-id</i>] [<i>detail</i>]	Displays information about neighbors, including device type, interface name, number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors of a specific interface or expand the display to provide more detailed information.
show cdp traffic	Displays Cisco Discovery Protocol counters, including the number of packets sent and received and checksum errors.



CHAPTER 39

Configuring Simple Network Management Protocol

- [Prerequisites for SNMP, on page 731](#)
- [Restrictions for SNMP, on page 733](#)
- [Information About SNMP, on page 733](#)
- [How to Configure SNMP, on page 738](#)
- [Monitoring SNMP Status, on page 752](#)
- [SNMP Examples, on page 752](#)

Prerequisites for SNMP

Supported SNMP Versions

This software release supports the following SNMP versions:

- **SNMPv1**—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- **SNMPv2C** replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the community-string-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic. It has these features:
 - **SNMPv2**—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
 - **SNMPv2C**—The community-string-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.
- **SNMPv3**—Version 3 of the SNMP is an interoperable standards-based protocol defined in RFCs 2273 to 2275. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network and includes these security features:
 - **Message integrity**—Ensures that a packet was not tampered with in transit.
 - **Authentication**—Determines that the message is from a valid source.
 - **Encryption**—Mixes the contents of a package to prevent it from being read by an unauthorized source.



Note To select encryption, enter the **priv** keyword.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval function and more detailed error message reporting to management stations. The bulk retrieval function retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2C report the error type.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy set up for a user and the group within which the user resides. A security level is the permitted level of security within a security model. A combination of the security level and the security model determine which security method is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

The following table identifies characteristics and compares different combinations of security models and levels:

Table 70: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	Result
SNMPv1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv2C	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv3	noAuthNoPriv	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.

Model	Level	Authentication	Encryption	Result
SNMPv3	authPriv	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	<p>Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.</p> <p>Allows specifying the User-based Security Model (USM) with these encryption algorithms:</p> <ul style="list-style-type: none"> • DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard. • 3DES 168-bit encryption • AES 128-bit, 192-bit, or 256-bit encryption

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

Restrictions for SNMP

Version Restrictions

- SNMPv1 does not support informs.

Information About SNMP

SNMP Overview

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a management information

base (MIB). The SNMP manager can be part of a network management system (NMS) such as Cisco Prime Infrastructure. The agent and MIB reside on the device. To configure SNMP on the device, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in the following table:

Table 71: SNMP Operations

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table. ³
get-bulk-request ⁴	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

³ With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.

⁴ The get-bulk command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the device, the community string definitions on the NMS must match at least one of the three community string definitions on the device.

A community string can have one of the following attributes:

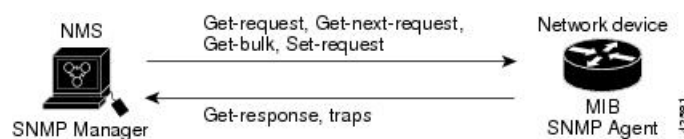
- Read-only (RO)—Gives all objects in the MIB except the community strings read access to authorized management stations, but does not allow write access.
- Read-write (RW)—Gives all objects in the MIB read and write access to authorized management stations, but does not allow access to the community strings.
- When a cluster is created, the command device manages the exchange of messages among member devices and the SNMP application. The Network Assistant software appends the member device number (@esN, where N is the device number) to the first configured RW and RO community strings on the command device and propagates them to the member devices.

SNMP MIB Variables Access

An example of an NMS is the Cisco Prime Infrastructure network management software. Cisco Prime Infrastructure software uses the device MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in the figure, the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in *get-request*, *get-next-request*, and *set-request* format.

Figure 65: SNMP Network



SNMP Notifications

SNMP allows the device to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or inform requests. In command syntax, unless there is an option in the command to select either traps or informs, the keyword traps refers to either traps or informs, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or informs.



Note SNMPv1 does not support informs.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be resent, informs are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the device and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be resent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the device is a concern and notification is not required, use traps.

SNMP ifIndex MIB Object Values

In an NMS, the IF-MIB generates and assigns an interface index (ifIndex) object value that is a unique number greater than zero to identify a physical or a logical interface. When the device reboots or the device software is upgraded, the device uses this same value for the interface. For example, if the device assigns a port 2 an ifIndex value of 10003, this value is the same after the device reboots.

The device uses one of the values in the following table to assign an ifIndex value to an interface:

Table 72: ifIndex Values

Interface Type	ifIndex Range
SVI ⁵	1–4999
EtherChannel	5001–5048
Tunnel	5078–5142
Physical (such as Gigabit Ethernet or SFP ⁶ -module interfaces) based on type and port numbers	10000–14500
Null	14501
Loopback and Tunnel	24567+

⁵ SVI = switch virtual interface

⁶ SFP = small form-factor pluggable

Default SNMP Configuration

Feature	Default Setting
SNMP agent	Disabled ⁷ .
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (tty).
SNMP version	If no version keyword is present, the default is Version 1.

Feature	Default Setting
SNMPv3 authentication	If no keyword is entered, the default is the noauth (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

⁷ This is the default when the device starts and the startup configuration does not have any **snmp-server** global configuration commands.

SNMP Configuration Guidelines

If the device starts and the device startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.

An SNMP *group* is a table that maps SNMP users to SNMP views. An SNMP *user* is a member of an SNMP group. An SNMP *host* is the recipient of an SNMP trap operation. An SNMP *engine ID* is a name for the local or remote SNMP engine.

When configuring SNMP, follow these guidelines:

- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command auto-generates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group.
- To configure a remote user, specify the IP address or port number for the remote SNMP agent of the device where the user resides.
- Before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineID** global configuration command with the **remote** option. The remote agent's SNMP engine ID and user password are used to compute the authentication and privacy digests. If you do not configure the remote engine ID first, the configuration command fails.
- When configuring SNMP informs, you need to configure the SNMP engine ID for the remote agent in the SNMP database before you can send proxy requests or informs to it.
- If a local user is not associated with a remote host, the device does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.
- Changing the value of the SNMP engine ID has significant results. A user's password (entered on the command line) is converted to an MD5 or SHA security digest based on the password and the local engine ID. The command-line password is then destroyed, as required by RFC 2274. Because of this deletion, if the value of the engine ID changes, the security digests of SNMPv3 users become invalid, and you need to reconfigure SNMP users by using the **snmp-server user username** global configuration command. Similar restrictions require the reconfiguration of community strings when the engine ID changes.

How to Configure SNMP

Disabling the SNMP Agent

The **no snmp-server** global configuration command disables all running versions (Version 1, Version 2C, and Version 3) of the SNMP agent on the device. You reenables all versions of the SNMP agent by the first **snmp-server** global configuration command that you enter. There is no Cisco IOS command specifically designated for enabling SNMP.

Follow these steps to disable the SNMP agent.

Before you begin

The SNMP Agent must be enabled before it can be disabled. The SNMP agent is enabled by the first **snmp-server** global configuration command entered on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no snmp-server**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	no snmp-server Example: Switch(config)# no snmp-server	Disables the SNMP agent operation.
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	<code>Switch(config)# end</code>	
Step 5	show running-config Example: <code>Switch# show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the device. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Follow these steps to configure a community string on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <code>Switch> enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>snmp-server community <i>string</i> [view <i>view-name</i>] [ro rw] [<i>access-list-number</i>]</p> <p>Example:</p> <pre>Switch(config)# snmp-server community comaccess ro 4</pre>	<p>Configures the community string.</p> <p>Note The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command.</p> <ul style="list-style-type: none"> • For <i>string</i>, specify a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. • (Optional) For view, specify the view record accessible to the community. • (Optional) Specify either read-only (ro) if you want authorized management stations to retrieve MIB objects, or specify read-write (rw) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. • (Optional) For <i>access-list-number</i>, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 4	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 4 deny any</pre>	<p>(Optional) If you specified an IP standard access list number in Step 3, then create the list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 3. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>source</i>, enter the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

To remove a specific community string, use the **no snmp-server** community string global configuration command.

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the device. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Configuring SNMP Groups and Users

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the device. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Follow these steps to configure SNMP groups and users on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server engineID** {local *engineid-string* | remote *ip-address* [**udp-port** *port-number*] *engineid-string*}
4. **snmp-server group** *group-name* {v1 | v2c | v3 {auth | noauth | priv}} [read *readview*] [write *writeview*] [notify *notifyview*] [access *access-list*]
5. **snmp-server user** *username* *group-name* {remote *host* [**udp-port** *port*]} {v1 [access *access-list*] | v2c [access *access-list*] | v3 [encrypted] [access *access-list*] [auth {md5 | sha} *auth-password*]} [priv {des | 3des | aes {128 | 192 | 256}} *priv-password*]

6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>snmp-server engineID {local <i>engineid-string</i> remote <i>ip-address</i> [<i>udp-port port-number</i>] <i>engineid-string</i>}</code></p> <p>Example:</p> <pre>Switch(config)# snmp-server engineID local 1234</pre>	<p>Configures a name for either the local or remote copy of SNMP.</p> <ul style="list-style-type: none"> • The <i>engineid-string</i> is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it has trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. The Step Example configures an engine ID of 123400000000000000000000. • If you select remote, specify the <i>ip-address</i> of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port on the remote device. The default is 162.
Step 4	<p><code>snmp-server group <i>group-name</i> {v1 v2c v3 {auth noauth priv}} [<i>read readview</i>] [<i>write writeview</i>] [<i>notify notifyview</i>] [<i>access access-list</i>]</code></p> <p>Example:</p> <pre>Switch(config)# snmp-server group public v2c access lmnop</pre>	<p>Configures a new SNMP group on the remote device.</p> <p>For <i>group-name</i>, specify the name of the group.</p> <p>Specify one of the following security models:</p> <ul style="list-style-type: none"> • v1 is the least secure of the possible security models. • v2c is the second least secure model. It allows transmission of informs and integers twice the normal width. • v3, the most secure, requires you to select one of the following authentication levels: <ul style="list-style-type: none"> auth—Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication.

	Command or Action	Purpose
		<p>noauth—Enables the noAuthNoPriv security level. This is the default if no keyword is specified.</p> <p>priv—Enables Data Encryption Standard (DES) packet encryption (also called privacy).</p> <p>(Optional) Enter read <i>readview</i> with a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent.</p> <p>(Optional) Enter write <i>writeview</i> with a string (not to exceed 64 characters) that is the name of the view in which you enter data and configure the contents of the agent.</p> <p>(Optional) Enter notify <i>notifyview</i> with a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap.</p> <p>(Optional) Enter access <i>access-list</i> with a string (not to exceed 64 characters) that is the name of the access list.</p>
<p>Step 5</p>	<p>snmp-server user <i>username group-name</i> { remote <i>host</i> [udp-port <i>port</i>] } { v1 [access <i>access-list</i>] v2c [access <i>access-list</i>] v3 [encrypted] [access <i>access-list</i>] [auth { md5 sha } <i>auth-password</i>] } [priv { des 3des aes { 128 192 256 } } <i>priv-password</i>]</p> <p>Example:</p> <pre>Switch(config)# snmp-server user Pat public v2c</pre>	<p>Adds a new user for an SNMP group.</p> <p>The <i>username</i> is the name of the user on the host that connects to the agent.</p> <p>The <i>group-name</i> is the name of the group to which the user is associated.</p> <p>Enter remote to specify a remote SNMP entity to which the user belongs and the hostname or IP address of that entity with the optional UDP port number. The default is 162.</p> <p>Enter the SNMP version number (v1, v2c, or v3). If you enter v3, you have these additional options:</p> <ul style="list-style-type: none"> • encrypted specifies that the password appears in encrypted format. This keyword is available only when the v3 keyword is specified. • auth is an authentication level setting session that can be either the HMAC-MD5-96 (md5) or the HMAC-SHA-96 (sha) authentication level and requires a password string <i>auth-password</i> (not to exceed 64 characters). <p>If you enter v3 you can also configure a private (priv) encryption algorithm and password string <i>priv-password</i> using the following keywords (not to exceed 64 characters):</p> <ul style="list-style-type: none"> • priv specifies the User-based Security Model (USM). • des specifies the use of the 56-bit DES algorithm. • 3des specifies the use of the 168-bit DES algorithm.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • aes specifies the use of the DES algorithm. You must select either 128-bit, 192-bit, or 256-bit encryption. (Optional) Enter access <i>access-list</i> with a string (not to exceed 64 characters) that is the name of the access list.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the device generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Devices running this Cisco IOS release can have an unlimited number of trap managers.



Note Many commands use the word **traps** in the command syntax. Unless there is an option in the command to select either traps or informs, the keyword **traps** refers to traps, informs, or both. Use the **snmp-server host** global configuration command to specify whether to send SNMP notifications as traps or informs.

You can use the **snmp-server host** global configuration command for a specific host to receive the notification types listed in the following table. You can enable any or all of these traps and configure a trap manager to receive them.

Table 73: Device Notification Types

Notification Type Keyword	Description
bridge	Generates STP bridge MIB traps.
cluster	Generates a trap when the cluster configuration changes.
config	Generates a trap for SNMP configuration changes.
copy-config	Generates a trap for SNMP copy configuration changes.

Notification Type Keyword	Description
cpu threshold	Allow CPU-related traps.
entity	Generates a trap for SNMP entity changes.
envmon	Generates environmental monitor traps. You can enable any or all of these environmental traps: fan, shutdown, status, supply, temperature.
errdisable	Generates a trap for a port VLAN errdisabled. You can also set a maximum trap rate per minute. The range is from 0 to 10000; the default is 0, which means there is no rate limit.
flash	Generates SNMP FLASH notifications. In a device stack, you can optionally enable notification for flash insertion or removal, which would cause a trap to be issued whenever a device in the stack is removed or inserted (physical removal, power cycle, or reload).
fru-ctrl	Generates entity field-replaceable unit (FRU) control traps. In the device stack, this trap refers to the insertion or removal of a device in the stack.
hsrp	Generates a trap for Hot Standby Router Protocol (HSRP) changes.
ipmulticast	Generates a trap for IP multicast routing changes.
ipsla	Generates a trap for the SNMP IP Service Level Agreements (SLAs).
mac-notification	Generates a trap for MAC address notifications.
msdp	Generates a trap for Multicast Source Discovery Protocol (MSDP) changes.
ospf	Generates a trap for Open Shortest Path First (OSPF) changes. You can enable any or all of these traps: Cisco specific, errors, link-state advertisement, rate limit, retransmit, and state changes.
pim	Generates a trap for Protocol-Independent Multicast (PIM) changes. You can enable any or all of these traps: invalid PIM messages, neighbor changes, and rendezvous point (RP)-mapping changes.
port-security	<p>Generates SNMP port security traps. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0, which means that there is no rate limit.</p> <p>Note When you configure a trap by using the notification type port-security, configure the port security trap first, and then configure the port security trap rate:</p> <ol style="list-style-type: none"> snmp-server enable traps port-security snmp-server enable traps port-security trap-rate rate
snmp	Generates a trap for SNMP-type notifications for authentication, cold start, warm start, link up or link down.

Notification Type Keyword	Description
storm-control	Generates a trap for SNMP storm-control. You can also set a maximum trap rate per minute. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vlancreate	Generates SNMP VLAN created traps.
vlandelete	Generates SNMP VLAN deleted traps.
vtp	Generates a trap for VLAN Trunking Protocol (VTP) changes.

Follow these steps to configure the device to send traps or informs to a host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server engineID remote** *ip-address engineid-string*
4. **snmp-server user** *username group-name* { **remote** *host* [**udp-port** *port*] } { **v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** { **md5** | **sha** } *auth-password*] }
5. **snmp-server group** *group-name* { **v1** | **v2c** | **v3** { **auth** | **noauth** | **priv** } } [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
6. **snmp-server host** *host-addr* [**informs** | **traps**] [**version** { **1** | **2c** | **3** { **auth** | **noauth** | **priv** } }] *community-string* [*notification-type*]
7. **snmp-server enable traps** *notification-types*
8. **snmp-server trap-source** *interface-id*
9. **snmp-server queue-length** *length*
10. **snmp-server trap-timeout** *seconds*
11. **end**
12. **show running-config**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>snmp-server engineID remote <i>ip-address engineid-string</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b</pre>	Specifies the engine ID for the remote host.
Step 4	<p>snmp-server user <i>username group-name</i> { remote host [udp-port <i>port</i>] } { v1 [access <i>access-list</i>] v2c [access <i>access-list</i>] v3 [encrypted] [access <i>access-list</i>] } [auth { md5 sha } <i>auth-password</i>] }</p> <p>Example:</p> <pre>Switch(config)# snmp-server user Pat public v2c</pre>	<p>Configures an SNMP user to be associated with the remote host created in Step 3.</p> <p>Note You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.</p>
Step 5	<p>snmp-server group <i>group-name</i> { v1 v2c v3 { auth noauth priv } } [read <i>readview</i>] [write <i>writeview</i>] [notify <i>notifyview</i>] [access <i>access-list</i>]</p> <p>Example:</p> <pre>Switch(config)# snmp-server group public v2c access lmnop</pre>	Configures an SNMP group.
Step 6	<p>snmp-server host <i>host-addr</i> [informs traps] [version { 1 2c 3 { auth noauth priv } }] <i>community-string</i> [<i>notification-type</i>]</p> <p>Example:</p> <pre>Switch(config)# snmp-server host 203.0.113.1 comaccess snmp</pre>	<p>Specifies the recipient of an SNMP trap operation.</p> <p>For <i>host-addr</i>, specify the name or Internet address of the host (the targeted recipient).</p> <p>(Optional) Specify traps (the default) to send SNMP traps to the host.</p> <p>(Optional) Specify informs to send SNMP informs to the host.</p> <p>(Optional) Specify the SNMP version (1, 2c, or 3). SNMPv1 does not support informs.</p> <p>(Optional) For Version 3, select authentication level auth, noauth, or priv.</p> <p>Note The priv keyword is available only when the cryptographic software image is installed.</p> <p>For <i>community-string</i>, when version 1 or version 2c is specified, enter the password-like community string sent with the notification operation. When version 3 is specified, enter the SNMPv3 username.</p>

	Command or Action	Purpose
		<p>The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command.</p> <p>(Optional) For <i>notification-type</i>, use the keywords listed in the table above. If no type is specified, all notifications are sent.</p>
Step 7	<p>snmp-server enable traps <i>notification-types</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server enable traps snmp</pre>	<p>Enables the device to send traps or informs and specifies the type of notifications to be sent. For a list of notification types, see the table above, or enter snmp-server enable traps ?</p> <p>To enable multiple types of traps, you must enter a separate snmp-server enable traps command for each trap type.</p> <p>Note When you configure a trap by using the notification type port-security, configure the port security trap first, and then configure the port security trap rate:</p> <ol style="list-style-type: none"> snmp-server enable traps port-security snmp-server enable traps port-security trap-rate rate
Step 8	<p>snmp-server trap-source <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server trap-source gigabitethernet 1/0/1</pre>	<p>(Optional) Specifies the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.</p>
Step 9	<p>snmp-server queue-length <i>length</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server queue-length 20</pre>	<p>(Optional) Establishes the message queue length for each trap host. The range is 1 to 5000; the default is 10.</p>
Step 10	<p>snmp-server trap-timeout <i>seconds</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server trap-timeout 60</pre>	<p>(Optional) Defines how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.</p>
Step 11	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 12	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 13	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

The **snmp-server host** command specifies which hosts receive the notifications. The **snmp-server enable traps** command globally enables the method for the specified notification (for traps and informs). To enable a host to receive an inform, you must configure an **snmp-server host informs** command for the host and globally enable informs by using the **snmp-server enable traps** command.

To remove the specified host from receiving traps, use the **no snmp-server host** *host* global configuration command. The **no snmp-server host** command with no keywords disables traps, but not informs, to the host. To disable informs, use the **no snmp-server host informs** global configuration command. To disable a specific trap type, use the **no snmp-server enable traps** *notification-types* global configuration command.

Setting the Agent Contact and Location Information

Follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server contact** *text*
4. **snmp-server location** *text*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	snmp-server contact <i>text</i> Example: Switch(config)# snmp-server contact Dial System Operator at beeper 21555	Sets the system contact string.
Step 4	snmp-server location <i>text</i> Example: Switch(config)# snmp-server location Building 3/Room 222	Sets the system location string.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Limiting TFTP Servers Used Through SNMP

Follow these steps to limit the TFTP servers used for saving and loading configuration files through SNMP to the servers specified in an access list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server tftp-server-list** *access-list-number*
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	snmp-server tftp-server-list access-list-number Example: Switch(config)# snmp-server tftp-server-list 44	Limits the TFTP servers used for configuration file copies through SNMP to the servers in the access list. For <i>access-list-number</i> , enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 4	access-list access-list-number {deny permit} source [source-wildcard] Example: Switch(config)# access-list 44 permit 10.1.1.2	Creates a standard access list, repeating the command as many times as necessary. For <i>access-list-number</i> , enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i> , enter the IP address of the TFTP servers that can access the device. (Optional) For <i>source-wildcard</i> , enter the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. You also can use the other privileged EXEC commands listed in the table to display SNMP information.

Table 74: Commands for Displaying SNMP Information

Command	Purpose
show snmp	Displays SNMP statistics.
show snmp group	Displays information on each SNMP group on the network.
show snmp pending	Displays information on pending SNMP requests.
show snmp sessions	Displays information on the current SNMP sessions.
show snmp user	Displays information on each SNMP user name in the SNMP users table. Note You must use this command to display SNMPv3 configuration information for auth noauth priv mode. This information is displayed in the show running-config output.

SNMP Examples

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the device to send any traps.

```
Switch(config)# snmp-server community public
```

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string *public*. The device also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Switch(config)# snmp-server community public
Switch(config)# snmp-server enable traps vtp
Switch(config)# snmp-server host 192.180.1.27 version 2c public
Switch(config)# snmp-server host 192.180.1.111 version 1 public
Switch(config)# snmp-server host 192.180.1.33 public
```


This example shows how to allow read-only access for all objects to members of access list 4 that use the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Switch(config)# snmp-server community comaccess ro 4
Switch(config)# snmp-server enable traps snmp authentication
Switch(config)# snmp-server host cisco.com version 2c public
```

This example shows how to send Entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the device to send Entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server** host commands for the host *cisco.com*.

```
Switch(config)# snmp-server enable traps entity
Switch(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the device to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host myhost.cisco.com public
```

This example shows how to associate a user with a remote host and to send **auth** (authNoPriv) authentication-level informs when the user enters global configuration mode:

```
Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Switch(config)# snmp-server group authgroup v3 auth
Switch(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5 mypassword
Switch(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Switch(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server inform retries 0
```




CHAPTER 40

Configuring SPAN and RSPAN

- [Prerequisites for SPAN and RSPAN, on page 755](#)
- [Restrictions for SPAN and RSPAN, on page 755](#)
- [Information About SPAN and RSPAN, on page 757](#)
- [How to Configure SPAN and RSPAN, on page 767](#)
- [Monitoring SPAN and RSPAN Operations, on page 789](#)
- [SPAN and RSPAN Configuration Examples, on page 789](#)

Prerequisites for SPAN and RSPAN

SPAN

- You can limit SPAN traffic to specific VLANs by using the **filter vlan** keyword. If a trunk port is being monitored, only traffic on the VLANs specified with this keyword is monitored. By default, all VLANs are monitored on a trunk port.

RSPAN

- We recommend that you configure an RSPAN VLAN before you configure an RSPAN source or a destination session.

Restrictions for SPAN and RSPAN

SPAN

The restrictions for SPAN are as follows:

- On each device, you can configure 66 sessions. A maximum of source sessions can be configured and the remaining sessions can be configured as RSPAN destinations sessions. A source session is either a local SPAN session or an RSPAN source session.
- For SPAN sources, you can monitor traffic for a single port or VLAN or a series or range of ports or VLANs for each session. You cannot mix source ports and source VLANs within a single SPAN session.
- The destination port cannot be a source port; a source port cannot be a destination port.

- You cannot have two SPAN sessions using the same destination port.
- When you configure a device port as a SPAN destination port, it is no longer a normal device port; only monitored traffic passes through the SPAN destination port.
- Entering SPAN configuration commands does not remove previously configured SPAN parameters. You must enter the **no monitor session** {*session_number* | **all** | **local** | **remote**} global configuration command to delete configured SPAN parameters.
- For local SPAN, outgoing packets through the SPAN destination port carry the original encapsulation headers—untagged, ISL, or IEEE 802.1Q—if the **encapsulation replicate** keywords are specified. If the keywords are not specified, the packets are sent in native form.
- You can configure a disabled port to be a source or destination port, but the SPAN function does not start until the destination port and at least one source port or source VLAN are enabled.
- You cannot mix source VLANs and filter VLANs within a single SPAN session.

Traffic monitoring in a SPAN session has the following restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- Wireshark does not capture egress packets when egress span is active.
- You can run both a local SPAN and an RSPAN source session in the same device or device stack. The device or device stack supports a total of 66 source and RSPAN destination sessions.
- You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.
- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports per device stack.
- SPAN sessions do not interfere with the normal operation of the device. However, an oversubscribed SPAN destination, for example, a 10-Mb/s port monitoring a 100-Mb/s port, can result in dropped or lost packets.
- When SPAN or RSPAN is enabled, each packet being monitored is sent twice, once as normal traffic and once as a monitored packet. Monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The device does not support a combination of local SPAN and RSPAN in a single session.
 - An RSPAN source session cannot have a local destination port.
 - An RSPAN destination session cannot have a local source port.
 - An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same device or device stack.

RSPAN

The restrictions for RSPAN are as follows:

- RSPAN does not support BPDU packet monitoring or other Layer 2 device protocols.
- The RSPAN VLAN is configured only on trunk ports and not on access ports. To avoid unwanted traffic in RSPAN VLANs, make sure that the VLAN remote-span feature is supported in all the participating devices.
- RSPAN VLANs are included as sources for port-based RSPAN sessions when source trunk ports have active RSPAN VLANs. RSPAN VLANs can also be sources in SPAN sessions. However, since the device does not monitor spanned traffic, it does not support egress spanning of packets on any RSPAN VLAN identified as the destination of an RSPAN source session on the device.
- CDP packets are not forwarded in RSPAN configured VLAN due to limitation in hardware. The workaround is to disable CDP on all the interfaces carrying RSPAN VLAN on the devices connected to the switch.
- If you enable VTP and VTP pruning, RSPAN traffic is pruned in the trunks to prevent the unwanted flooding of RSPAN traffic across the network for VLAN IDs that are lower than 1005.
- To use RSPAN, the switch must be running the LAN Base image.

Information About SPAN and RSPAN

SPAN and RSPAN

You can analyze network traffic passing through ports or VLANs by using SPAN or RSPAN to send a copy of the traffic to another port on the device or on another device that has been connected to a network analyzer or other monitoring or security device. SPAN copies (or mirrors) traffic received or sent (or both) on source ports or source VLANs to a destination port for analysis. SPAN does not affect the switching of network traffic on the source ports or VLANs. You must dedicate the destination port for SPAN use. Except for traffic that is required for the SPAN or RSPAN session, destination ports do not receive or forward traffic.

Only traffic that enters or leaves source ports or traffic that enters or leaves source VLANs can be monitored by using SPAN; traffic routed to a source VLAN cannot be monitored. For example, if incoming traffic is being monitored, traffic that gets routed from another VLAN to the source VLAN cannot be monitored; however, traffic that is received on the source VLAN and routed to another VLAN can be monitored.

You can use the SPAN or RSPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) sensor appliance to a destination port, the IDS device can send TCP reset packets to close down the TCP session of a suspected attacker.

Local SPAN

Local SPAN supports a SPAN session entirely within one device; all source ports or source VLANs and destination ports are in the same device or device stack. Local SPAN copies traffic from one or more source ports in any VLAN or from one or more VLANs to a destination port for analysis.

Figure 66: Example of Local SPAN Configuration on a Single Device

All traffic on port 5 (the source port) is mirrored to port 10 (the destination port). A network analyzer on port 10 receives all network traffic from port 5 without being physically attached to port

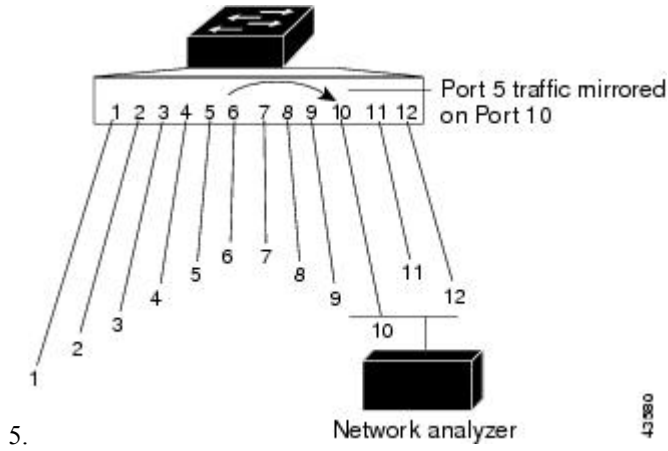
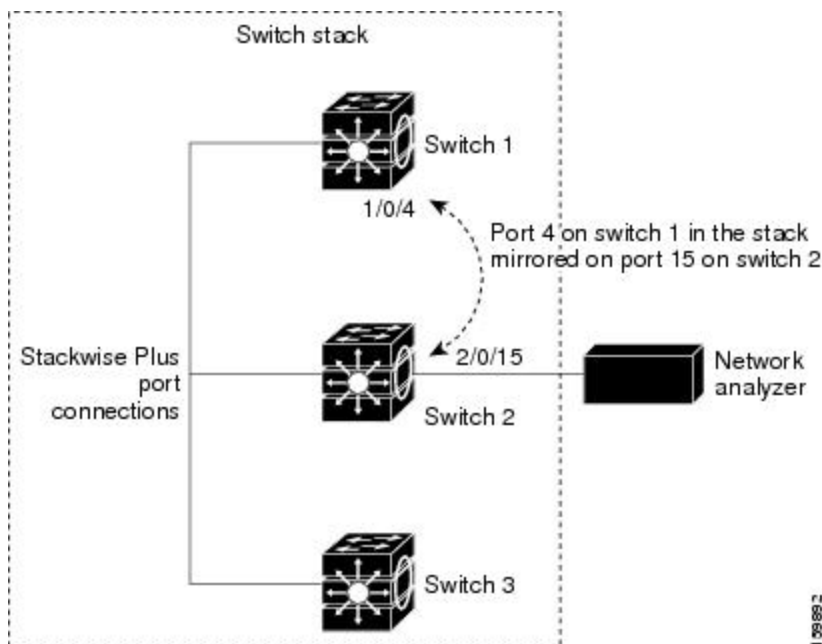


Figure 67: Example of Local SPAN Configuration on a Device Stack

This is an example of a local SPAN in a device stack, where the source and destination ports reside on different stack members.



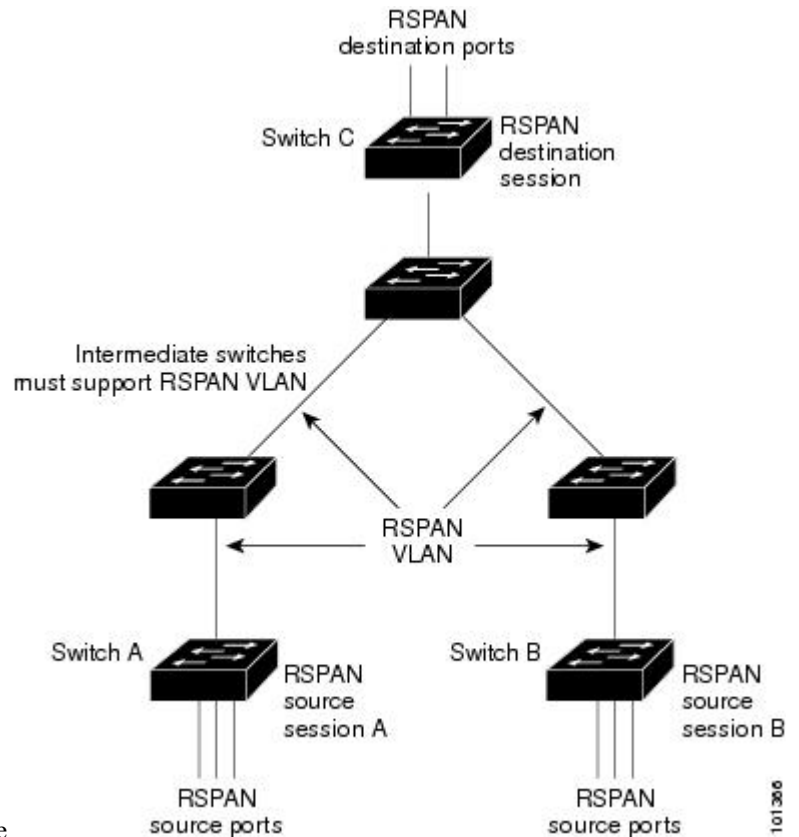
Remote SPAN

RSPAN supports source ports, source VLANs, and destination ports on different devices (or different device stacks), enabling remote monitoring of multiple devices across your network.

Figure 68: Example of RSPAN Configuration

The figure below shows source ports on Device A and Device B. The traffic for each RSPAN session is carried over a user-specified RSPAN VLAN that is dedicated for that RSPAN session in all participating devices.

The RSPAN traffic from the source ports or VLANs is copied into the RSPAN VLAN and forwarded over trunk ports carrying the RSPAN VLAN to a destination session monitoring the RSPAN VLAN. Each RSPAN source device must have either ports or VLANs as RSPAN sources. The destination is always a physical port,



as shown on Device C in the figure.

SPAN and RSPAN Concepts and Terminology

SPAN Sessions

SPAN sessions (local or remote) allow you to monitor traffic on one or more ports, or one or more VLANs, and send the monitored traffic to one or more destination ports.

A local SPAN session is an association of a destination port with source ports or source VLANs, all on a single network device. Local SPAN does not have separate source and destination sessions. Local SPAN sessions gather a set of ingress and egress packets specified by the user and form them into a stream of SPAN data, which is directed to the destination port.

RSPAN consists of at least one RSPAN source session, an RSPAN VLAN, and at least one RSPAN destination session. You separately configure RSPAN source sessions and RSPAN destination sessions on different network devices. To configure an RSPAN source session on a device, you associate a set of source ports or source VLANs with an RSPAN VLAN. The output of this session is the stream of SPAN packets that are sent to the RSPAN VLAN. To configure an RSPAN destination session on another device, you associate the destination port with the RSPAN VLAN. The destination session collects all RSPAN VLAN traffic and sends it out the RSPAN destination port.

An RSPAN source session is very similar to a local SPAN session, except for where the packet stream is directed. In an RSPAN source session, SPAN packets are relabeled with the RSPAN VLAN ID and directed over normal trunk ports to the destination device.

An RSPAN destination session takes all packets received on the RSPAN VLAN, strips off the VLAN tagging, and presents them on the destination port. The session presents a copy of all RSPAN VLAN packets (except Layer 2 control packets) to the user for analysis.

Traffic monitoring in a SPAN session has these restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- You can run both a local SPAN and an RSPAN source session in the same device or device stack. The device or device stack supports a total of 66 source and RSPAN destination sessions.
- You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.
- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports per device stack.
- SPAN sessions do not interfere with the normal operation of the device. However, an oversubscribed SPAN destination, for example, a 10-Mb/s port monitoring a 100-Mb/s port, can result in dropped or lost packets.
- When SPAN or RSPAN is enabled, each packet being monitored is sent twice, once as normal traffic and once as a monitored packet. Therefore monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The device does not support a combination of local SPAN and RSPAN in a single session.
 - An RSPAN source session cannot have a local destination port.
 - An RSPAN destination session cannot have a local source port.
 - An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same device or device stack.

Monitored Traffic

SPAN sessions can monitor these traffic types:

- Receive (Rx) SPAN—Receive (or ingress) SPAN monitors as much as possible all of the packets received by the source interface or VLAN before any modification or processing is performed by the device. A copy of each packet received by the source is sent to the destination port for that SPAN session.

Packets that are modified because of routing or Quality of Service (QoS)—for example, modified Differentiated Services Code Point (DSCP)—are copied before modification.

Features that can cause a packet to be dropped during receive processing have no effect on ingress SPAN; the destination port receives a copy of the packet even if the actual incoming packet is dropped. These features include IP standard and extended input Access Control Lists (ACLs), ingress QoS policing, VLAN ACLs, and egress QoS policing.

- Transmit (Tx) SPAN—Transmit (or egress) SPAN monitors as much as possible all of the packets sent by the source interface after all modification and processing is performed by the device. A copy of each packet sent by the source is sent to the destination port for that SPAN session. The copy is provided after the packet is modified.

Packets that are modified because of routing (for example, with modified time-to-live (TTL), MAC address, or QoS values) are duplicated (with the modifications) at the destination port.

Features that can cause a packet to be dropped during transmit processing also affect the duplicated copy for SPAN. These features include IP standard and extended output ACLs and egress QoS policing.

- Both—In a SPAN session, you can also monitor a port or VLAN for both received and sent packets. This is the default.

Therefore, a local SPAN session with encapsulation replicate enabled can have a mixture of untagged and IEEE 802.1Q tagged packets appear on the destination port.

Device congestion can cause packets to be dropped at ingress source ports, egress source ports, or SPAN destination ports. In general, these characteristics are independent of one another. For example:

- A packet might be forwarded normally but dropped from monitoring due to an oversubscribed SPAN destination port.
- An ingress packet might be dropped from normal forwarding, but still appear on the SPAN destination port.
- An egress packet dropped because of device congestion is also dropped from egress SPAN.

In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination port. For example, a bidirectional (both Rx and Tx) SPAN session is configured for the Rx monitor on port A and Tx monitor on port B. If a packet enters the device through port A and is switched to port B, both incoming and outgoing packets are sent to the destination port. Both packets are the same unless a Layer 3 rewrite occurs, in which case the packets are different because of the packet modification.

Source Ports

A source port (also called a monitored port) is a switched or routed port that you monitor for network traffic analysis.

In a local SPAN session or RSPAN source session, you can monitor source ports or VLANs for traffic in one or both directions.

The device supports any number of source ports (up to the maximum number of available ports on the device) and any number of source VLANs (up to the maximum number of VLANs supported).

However, the device supports a maximum of four sessions (two sessions if device is in a stack with Catalyst 2960-S switches) (local or RSPAN) with source ports or VLANs. You cannot mix ports and VLANs in a single session.

A source port has these characteristics:

- It can be monitored in multiple SPAN sessions.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor.
- It can be any port type (for example, EtherChannel, Gigabit Ethernet, and so forth).
- For EtherChannel sources, you can monitor traffic for the entire EtherChannel or individually on a physical port as it participates in the port channel.
- It can be an access port, trunk port, routed port, or voice VLAN port.
- It cannot be a destination port.

- Source ports can be in the same or different VLANs.
- You can monitor multiple source ports in a single session.

Source VLANs

VLAN-based SPAN (VSPAN) is the monitoring of the network traffic in one or more VLANs. The SPAN or RSPAN source interface in VSPAN is a VLAN ID, and traffic is monitored on all the ports for that VLAN.

VSPAN has these characteristics:

- All active ports in the source VLAN are included as source ports and can be monitored in either or both directions.
- On a given port, only traffic on the monitored VLAN is sent to the destination port.
- If a destination port belongs to a source VLAN, it is excluded from the source list and is not monitored.
- If ports are added to or removed from the source VLANs, the traffic on the source VLAN received by those ports is added to or removed from the sources being monitored.
- You cannot use filter VLANs in the same session with VLAN sources.
- You can monitor only Ethernet VLANs.

VLAN Filtering

When you monitor a trunk port as a source port, by default, all VLANs active on the trunk are monitored. You can limit SPAN traffic monitoring on trunk source ports to specific VLANs by using VLAN filtering.

- VLAN filtering applies only to trunk ports or to voice VLAN ports.
- VLAN filtering applies only to port-based sessions and is not allowed in sessions with VLAN sources.
- When a VLAN filter list is specified, only those VLANs in the list are monitored on trunk ports or on voice VLAN access ports.
- SPAN traffic coming from other port types is not affected by VLAN filtering; that is, all VLANs are allowed on other ports.
- VLAN filtering affects only traffic forwarded to the destination SPAN port and does not affect the switching of normal traffic.

Destination Port

Each local SPAN session or RSPAN destination session must have a destination port (also called a monitoring port) that receives a copy of traffic from the source ports or VLANs and sends the SPAN packets to the user, usually a network analyzer.

A destination port has these characteristics:

- For a local SPAN session, the destination port must reside on the same device or device stack as the source port. For an RSPAN session, it is located on the device containing the RSPAN destination session. There is no destination port on a device or device stack running only an RSPAN source session.
- When a port is configured as a SPAN destination port, the configuration overwrites the original port configuration. When the SPAN destination configuration is removed, the port reverts to its previous

configuration. If a configuration change is made to the port while it is acting as a SPAN destination port, the change does not take effect until the SPAN destination configuration had been removed.

- If the port was in an EtherChannel group, it is removed from the group while it is a destination port. If it was a routed port, it is no longer a routed port.
- It can be any Ethernet physical port.
- It cannot be a secure port.
- It cannot be a source port.
- It can participate in only one SPAN session at a time (a destination port in one SPAN session cannot be a destination port for a second SPAN session).
- When it is active, incoming traffic is disabled. The port does not transmit any traffic except that required for the SPAN session. Incoming traffic is never learned or forwarded on a destination port.
- If ingress traffic forwarding is enabled for a network security device, the destination port forwards traffic at Layer 2.
- It does not participate in any of the Layer 2 protocols (STP, VTP, CDP, DTP, PagP).
- A destination port that belongs to a source VLAN of any SPAN session is excluded from the source list and is not monitored.
- The maximum number of destination ports in a device or device stack is 64.

Local SPAN and RSPAN destination ports function differently with VLAN tagging and encapsulation:

- For local SPAN, if the **encapsulation replicate** keywords are specified for the destination port, these packets appear with the original encapsulation (untagged, ISL, or IEEE 802.1Q). If these keywords are not specified, packets appear in the untagged format. Therefore, the output of a local SPAN session with **encapsulation replicate** enabled can contain a mixture of untagged, ISL, or IEEE 802.1Q-tagged packets.
- For RSPAN, the original VLAN ID is lost because it is overwritten by the RSPAN VLAN identification. Therefore, all packets appear on the destination port as untagged.

RSPAN VLAN

The RSPAN VLAN carries SPAN traffic between RSPAN source and destination sessions. RSPAN VLAN has these special characteristics:

- All traffic in the RSPAN VLAN is always flooded.
- No MAC address learning occurs on the RSPAN VLAN.
- RSPAN VLAN traffic only flows on trunk ports.
- RSPAN VLANs must be configured in VLAN configuration mode by using the **remote-span** VLAN configuration mode command.
- STP can run on RSPAN VLAN trunks but not on SPAN destination ports.
- An RSPAN VLAN cannot be a private-VLAN primary or secondary VLAN.

For VLANs 1 to 1005 that are visible to VLAN Trunking Protocol (VTP), the VLAN ID and its associated RSPAN characteristic are propagated by VTP. If you assign an RSPAN VLAN ID in the extended VLAN range (1006 to 4094), you must manually configure all intermediate devices.

It is normal to have multiple RSPAN VLANs in a network at the same time with each RSPAN VLAN defining a network-wide RSPAN session. That is, multiple RSPAN source sessions anywhere in the network can contribute packets to the RSPAN session. It is also possible to have multiple RSPAN destination sessions throughout the network, monitoring the same RSPAN VLAN and presenting traffic to the user. The RSPAN VLAN ID separates the sessions.

SPAN and RSPAN Interaction with Other Features

SPAN interacts with these features:

- **Routing**—SPAN does not monitor routed traffic. VSPAN only monitors traffic that enters or exits the device, not traffic that is routed between VLANs. For example, if a VLAN is being Rx-monitored and the device routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and not received on the SPAN destination port.
- **STP**—A destination port does not participate in STP while its SPAN or RSPAN session is active. The destination port can participate in STP after the SPAN or RSPAN session is disabled. On a source port, SPAN does not affect the STP status. STP can be active on trunk ports carrying an RSPAN VLAN.
- **CDP**—A SPAN destination port does not participate in CDP while the SPAN session is active. After the SPAN session is disabled, the port again participates in CDP.
- **VTP**—You can use VTP to prune an RSPAN VLAN between devices.
- **VLAN and trunking**—You can modify VLAN membership or trunk settings for source or destination ports at any time. However, changes in VLAN membership or trunk settings for a destination port do not take effect until you remove the SPAN destination configuration. Changes in VLAN membership or trunk settings for a source port immediately take effect, and the respective SPAN sessions automatically adjust accordingly.
- **EtherChannel**—You can configure an EtherChannel group as a source port. When a group is configured as a SPAN source, the entire group is monitored.

If a physical port is added to a monitored EtherChannel group, the new port is added to the SPAN source port list. If a port is removed from a monitored EtherChannel group, it is automatically removed from the source port list.

A physical port that belongs to an EtherChannel group can be configured as a SPAN source port and still be a part of the EtherChannel. In this case, data from the physical port is monitored as it participates in the EtherChannel. However, if a physical port that belongs to an EtherChannel group is configured as a SPAN destination, it is removed from the group. After the port is removed from the SPAN session, it rejoins the EtherChannel group. Ports removed from an EtherChannel group remain members of the group, but they are in the inactive or suspended state.

If a physical port that belongs to an EtherChannel group is a destination port and the EtherChannel group is a source, the port is removed from the EtherChannel group and from the list of monitored ports.

- Multicast traffic can be monitored. For egress and ingress port monitoring, only a single unedited packet is sent to the SPAN destination port. It does not reflect the number of times the multicast packet is sent.
- A private-VLAN port cannot be a SPAN destination port.
- A secure port cannot be a SPAN destination port.

For SPAN sessions, do not enable port security on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable port security on any ports with monitored egress.

- An IEEE 802.1x port can be a SPAN source port. You can enable IEEE 802.1x on a port that is a SPAN destination port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination.

For SPAN sessions, do not enable IEEE 802.1x on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable IEEE 802.1x on any ports that are egress monitored.

Flow-Based SPAN

You can control the type of network traffic to be monitored in SPAN or RSPAN sessions by using flow-based SPAN (FSPAN) or flow-based RSPAN (FRSPAN), which apply access control lists (ACLs) to the monitored traffic on the source ports. The FSPAN ACLs can be configured to filter IPv4, IPv6, and non-IP monitored traffic.

You apply an ACL to a SPAN session through the interface. It is applied to all the traffic that is monitored on all interfaces in the SPAN session. The packets that are permitted by this ACL are copied to the SPAN destination port. No other packets are copied to the SPAN destination port.

The original traffic continues to be forwarded, and any port, VLAN, and router ACLs attached are applied. The FSPAN ACL does not have any effect on the forwarding decisions. Similarly, the port, VLAN, and router ACLs do not have any effect on the traffic monitoring. If a security input ACL denies a packet and it is not forwarded, the packet is still copied to the SPAN destination ports if the FSPAN ACL permits it. But if the security output ACL denies a packet and it is not sent, it is not copied to the SPAN destination ports. However, if the security output ACL permits the packet to go out, it is only copied to the SPAN destination ports if the FSPAN ACL permits it. This is also true for an RSPAN session.



Note When you configure an FSPAN session, ensure that you remove the existing SPAN sessions, configure the FSPAN session, and then reconfigure the SPAN sessions.

You can attach three types of FSPAN ACLs to the SPAN session:

- IPv4 FSPAN ACL— Filters only IPv4 packets.
- IPv6 FSPAN ACL— Filters only IPv6 packets.
- MAC FSPAN ACL— Filters only non-IP packets.

The security ACLs have higher priority than the FSPAN ACLs on a device. If FSPAN ACLs are applied, and you later add more security ACLs that cannot fit in the hardware memory, the FSPAN ACLs that you applied are removed from memory to allow space for the security ACLs. A system message notifies you of this action, which is called unloading. When there is again space for the FSPAN ACLs to reside in memory, they are added to the hardware memory on the device. A system message notifies you of this action, which is called reloading. The IPv4, IPv6 and MAC FSPAN ACLs can be unloaded or reloaded independently.

If a VLAN-based FSPAN session configured on a stack cannot fit in the hardware memory on one or more devices, it is treated as unloaded on those devices, and traffic meant for the FSPAN ACL and sourcing on that device is not copied to the SPAN destination ports. The FSPAN ACL continues to be correctly applied,

and traffic is copied to the SPAN destination ports on the devices where the FSPAN ACL fits in the hardware memory.

When an empty FSPAN ACL is attached, some hardware functions copy all traffic to the SPAN destination ports for that ACL. If sufficient hardware resources are not available, even an empty FSPAN ACL can be unloaded.

IPv4 and MAC FSPAN ACLs are supported on all feature sets. IPv6 FSPAN ACLs are supported only in the advanced IP Services feature set.

Default SPAN and RSPAN Configuration

Table 75: Default SPAN and RSPAN Configuration

Feature	Default Setting
SPAN state (SPAN and RSPAN)	Disabled.
Source port traffic to monitor	Both received and sent traffic (both).
Encapsulation type (destination port)	Native form (untagged packets).
Ingress forwarding (destination port)	Disabled.
VLAN filtering	On a trunk interface used as a source port, all VLANs are monitored.
RSPAN VLANs	None configured.

Configuration Guidelines

SPAN Configuration Guidelines

- To remove a source or destination port or VLAN from the SPAN session, use the **no monitor session session_number source {interface interface-id | vlan vlan-id}** global configuration command or the **no monitor session session_number destination interface interface-id** global configuration command. For destination interfaces, the **encapsulation** options are ignored with the **no** form of the command.
- To monitor all VLANs on the trunk port, use the **no monitor session session_number filter** global configuration command.

RSPAN Configuration Guidelines

- All the SPAN configuration guidelines apply to RSPAN.
- As RSPAN VLANs have special properties, you should reserve a few VLANs across your network for use as RSPAN VLANs; do not assign access ports to these VLANs.
- You can apply an output ACL to RSPAN traffic to selectively filter or monitor specific packets. Specify these ACLs on the RSPAN VLAN in the RSPAN source devices.

- For RSPAN configuration, you can distribute the source ports and the destination ports across multiple devices in your network.
- Access ports (including voice VLAN ports) on the RSPAN VLAN are put in the inactive state.
- You can configure any VLAN as an RSPAN VLAN as long as these conditions are met:
 - The same RSPAN VLAN is used for an RSPAN session in all the devices.
 - All participating devices support RSPAN.

FSPAN and FRSPAN Configuration Guidelines

- When at least one FSPAN ACL is attached, FSPAN is enabled.
- When you attach at least one FSPAN ACL that is not empty to a SPAN session, and you have not attached one or more of the other FSPAN ACLs (for instance, you have attached an IPv4 ACL that is not empty, and have not attached IPv6 and MAC ACLs), FSPAN blocks the traffic that would have been filtered by the unattached ACLs. Therefore, this traffic is not monitored.

How to Configure SPAN and RSPAN

Creating a Local SPAN Session

Follow these steps to create a SPAN session and specify the source (monitored) ports or VLANs and the destination (monitoring) ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [, | -] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation replicate**]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>no monitor session {<i>session_number</i> all local remote}</p> <p>Example:</p> <pre>Switch(config)# no monitor session all</pre>	<p>Removes any existing SPAN configuration for the session.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 4. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	<p>monitor session <i>session_number</i> source {interface <i>interface-id</i> vlan <i>vlan-id</i>} [, -] [both rx tx]</p> <p>Example:</p> <pre>Switch(config)# monitor session 1 source interface gigabitethernet1/0/1</pre>	<p>Specifies the SPAN session and the source port (monitored port).</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 4. • For <i>interface-id</i>, specify the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>). Valid port-channel numbers are 1 to 6. • For <i>vlan-id</i>, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). <p>Note A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session.</p> <ul style="list-style-type: none"> • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) both rx tx—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. <ul style="list-style-type: none"> • both—Monitors both received and sent traffic. • rx—Monitors received traffic. • tx—Monitors sent traffic.

	Command or Action	Purpose
		<p>Note You can use the monitor session session_number source command multiple times to configure multiple source ports.</p>
Step 5	<p>monitor session session_number destination {interface interface-id [, -] [encapsulation replicate]}</p> <p>Example:</p> <pre>Switch(config)# monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate</pre>	<p>Specifies the SPAN session and the destination port (monitoring port). The port LED changes to amber when the configuration changes take effect. The LED returns to its original state(green) only after removing the SPAN destination configuration.</p> <p>Note For local SPAN, you must use the same session number for the source and destination interfaces.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in step 4. • For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. <p>(Optional) encapsulation replicate specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).</p> <p>Note You can use monitor session session_number destination command multiple times to configure multiple destination ports.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Creating a Local SPAN Session and Configuring Incoming Traffic

Follow these steps to create a SPAN session, to specify the source ports or VLANs and the destination ports, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `no monitor session {session_number | all | local | remote}`
4. `monitor session session_number source {interface interface-id | vlan vlan-id} [, | -] [both | rx | tx]`
5. `monitor session session_number destination {interface interface-id [, | -] [encapsulation replicate[ingress {dot1q vlan vlan-id | untagged vlan vlan-id | vlan vlan-id}]}`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>no monitor session {session_number all local remote}</code></p> <p>Example:</p> <pre>Switch(config)# no monitor session all</pre>	<p>Removes any existing SPAN configuration for the session.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 4. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.

	Command or Action	Purpose
Step 4	<p>monitor session <i>session_number</i> source {interface <i>interface-id</i> / vlan <i>vlan-id</i>} [, -] [both rx tx]</p> <p>Example:</p> <pre>Switch(config)# monitor session 2 source gigabitethernet0/1 rx</pre>	Specifies the SPAN session and the source port (monitored port).
Step 5	<p>monitor session <i>session_number</i> destination {interface <i>interface-id</i> [, -] [encapsulation replicate[ingress {dot1q vlan <i>vlan-id</i> untagged vlan <i>vlan-id</i> vlan <i>vlan-id</i>}]}</p> <p>Example:</p> <pre>Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate ingress dot1q vlan 6</pre>	<p>Specifies the SPAN session, the destination port, the packet encapsulation, and the ingress VLAN and encapsulation.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [, -]—Specifies a series or range of interfaces. Enter a space before and after the comma or hyphen. • (Optional) encapsulation replicate—Specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). • ingress—Enables forwarding of incoming traffic on the destination port and to specify the encapsulation type. <ul style="list-style-type: none"> • dot1q vlan <i>vlan-id</i>—Accepts incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. • untagged vlan <i>vlan-id</i> or vlan <i>vlan-id</i>—Accepts incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.

	Command or Action	Purpose
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Specifying VLANs to Filter

Follow these steps to limit SPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [, | -]
6. **monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation replicate**]}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: <pre>Switch(config)# no monitor session all</pre>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.

	Command or Action	Purpose
Step 4	<p>monitor session <i>session_number</i> source interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx</pre>	<p>Specifies the characteristics of the source port (monitored port) and SPAN session.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>interface-id</i>, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 5	<p>monitor session <i>session_number</i> filter vlan <i>vlan-id</i> [, -]</p> <p>Example:</p> <pre>Switch(config)# monitor session 2 filter vlan 1 - 5 , 9</pre>	<p>Limits the SPAN source traffic to specific VLANs.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, enter the session number specified in Step 4. • For <i>vlan-id</i>, the range is 1 to 4094. • (Optional) Use a comma (,) to specify a series of VLANs, or use a hyphen (-) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 6	<p>monitor session <i>session_number</i> destination {interface <i>interface-id</i> [, -] [encapsulation replicate]}</p> <p>Example:</p> <pre>Switch(config)# monitor session 2 destination interface gigabitethernet1/0/1</pre>	<p>Specifies the SPAN session and the destination port (monitoring port).</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) encapsulation replicate specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).
Step 7	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a VLAN as an RSPAN VLAN

Follow these steps to create a new VLAN, then configure it to be the RSPAN VLAN for the RSPAN session.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan *vlan-id***
4. **remote-span**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	vlan <i>vlan-id</i> Example: <pre>Switch(config)# vlan 100</pre>	Enters a VLAN ID to create a VLAN, or enters the VLAN ID of an existing VLAN, and enters VLAN configuration mode. The range is 2 to 1001 and 1006 to 4094. The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved for Token Ring and FDDI VLANs).
Step 4	remote-span Example: <pre>Switch(config-vlan)# remote-span</pre>	Configures the VLAN as an RSPAN VLAN.

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config-vlan)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

You must create the RSPAN VLAN in all devices that will participate in RSPAN. If the RSPAN VLAN-ID is in the normal range (lower than 1005) and VTP is enabled in the network, you can create the RSPAN VLAN in one device, and VTP propagates it to the other devices in the VTP domain. For extended-range VLANs (greater than 1005), you must configure RSPAN VLAN on both source and destination devices and any intermediate devices.

Use VTP pruning to get an efficient flow of RSPAN traffic, or manually delete the RSPAN VLAN from all trunks that do not need to carry the RSPAN traffic.

To remove the remote SPAN characteristic from a VLAN and convert it back to a normal VLAN, use the **no remote-span** VLAN configuration command.

To remove a source port or VLAN from the SPAN session, use the **no monitor session *session_number* source {interface *interface-id* | vlan *vlan-id*}** global configuration command. To remove the RSPAN VLAN from the session, use the **no monitor session *session_number* destination remote vlan *vlan-id***.

Creating an RSPAN Source Session

Follow these steps to create and start an RSPAN source session and to specify the monitored source and the destination RSPAN VLAN.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session {*session_number* | all | local | remote}**
4. **monitor session *session_number* source {interface *interface-id* | vlan *vlan-id*} [, | -] [both | rx | tx]**
5. **monitor session *session_number* destination remote vlan *vlan-id***
6. **end**
7. **show running-config**

8. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: <pre>Switch(config)# no monitor session 1</pre>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source { interface <i>interface-id</i> vlan <i>vlan-id</i> } [, -] [both rx tx] Example: <pre>Switch(config)# monitor session 1 source interface gigabitethernet1/0/1 tx</pre>	Specifies the RSPAN session and the source port (monitored port). <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • Enter a source port or source VLAN for the RSPAN session: <ul style="list-style-type: none"> • For <i>interface-id</i>, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>). Valid port-channel numbers are 1 to 48. • For <i>vlan-id</i>, specifies the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). • A single session can include multiple sources (ports or VLANs), defined in a series of commands, but you cannot combine source ports and source VLANs in one session. • (Optional) [, -]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) both rx tx—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. <ul style="list-style-type: none"> • both—Monitors both received and sent traffic. • rx—Monitors received traffic. • tx—Monitors sent traffic.
Step 5	monitor session <i>session_number</i> destination remote vlan <i>vlan-id</i> Example: <pre>Switch(config)# monitor session 1 destination remote vlan 100</pre>	Specifies the RSPAN session, the destination RSPAN VLAN, and the destination-port group. <ul style="list-style-type: none"> • For <i>session_number</i>, enter the number defined in Step 4. • For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Specifying VLANs to Filter

Follow these steps to configure the RSPAN source session to limit RSPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [, | -]

6. **monitor session** *session_number* **destination remote vlan** *vlan-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: <pre>Switch(config)# no monitor session 2</pre>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source interface <i>interface-id</i> Example: <pre>Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx</pre>	Specifies the characteristics of the source port (monitored port) and SPAN session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>interface-id</i>, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 5	monitor session <i>session_number</i> filter vlan <i>vlan-id</i> [, -] Example: <pre>Switch(config)# monitor session 2 filter vlan 1 - 5 , 9</pre>	Limits the SPAN source traffic to specific VLANs. <ul style="list-style-type: none"> • For <i>session_number</i>, enter the session number specified in step 4. • For <i>vlan-id</i>, the range is 1 to 4094. • (Optional) , - Use a comma (,) to specify a series of VLANs or use a hyphen (-) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 6	monitor session <i>session_number</i> destination remote vlan <i>vlan-id</i>	Specifies the RSPAN session and the destination remote VLAN (RSPAN VLAN).

	Command or Action	Purpose
	Example: <pre>Switch(config)# monitor session 2 destination remote vlan 902</pre>	<ul style="list-style-type: none"> For <i>session_number</i>, enter the session number specified in Step 4. For <i>vlan-id</i>, specify the RSPAN VLAN to carry the monitored traffic to the destination port.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating an RSPAN Destination Session

You configure an RSPAN destination session on a different device or device stack; that is, not the device or device stack on which the source session was configured.

Follow these steps to define the RSPAN VLAN on that device, to create an RSPAN destination session, and to specify the source RSPAN VLAN and the destination port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*
4. **remote-span**
5. **exit**
6. **no monitor session** {*session_number* | **all** | **local** | **remote**}
7. **monitor session** *session_number* **source remote vlan** *vlan-id*
8. **monitor session** *session_number* **destination interface** *interface-id*
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vlan <i>vlan-id</i> Example: Switch(config)# vlan 901	Specifies the VLAN ID of the RSPAN VLAN created from the source device, and enters VLAN configuration mode. If both devices are participating in VTP and the RSPAN VLAN ID is from 2 to 1005, Steps 3 through 5 are not required because the RSPAN VLAN ID is propagated through the VTP network.
Step 4	remote-span Example: Switch(config-vlan)# remote-span	Identifies the VLAN as the RSPAN VLAN.
Step 5	exit Example: Switch(config-vlan)# exit	Returns to global configuration mode.
Step 6	no monitor session {<i>session_number</i> all local remote} Example: Switch(config)# no monitor session 1	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 7	monitor session <i>session_number</i> source remote vlan <i>vlan-id</i> Example: Switch(config)# monitor session 1 source remote vlan 901	Specifies the RSPAN session and the source RSPAN VLAN. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.

	Command or Action	Purpose
Step 8	<p>monitor session <i>session_number</i> destination interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# monitor session 1 destination interface gigabitethernet2/0/1</pre>	<p>Specifies the RSPAN session and the destination interface.</p> <ul style="list-style-type: none"> For <i>session_number</i>, enter the number defined in Step 7. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. For <i>interface-id</i>, specify the destination interface. The destination interface must be a physical interface. Though visible in the command-line help string, encapsulation replicate is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged.
Step 9	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating an RSPAN Destination Session and Configuring Incoming Traffic

Follow these steps to create an RSPAN destination session, to specify the source RSPAN VLAN and the destination port, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source remote vlan** *vlan-id*

5. **monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: <pre>Switch(config)# no monitor session 2</pre>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source remote vlan <i>vlan-id</i> Example: <pre>Switch(config)# monitor session 2 source remote vlan 901</pre>	Specifies the RSPAN session and the source RSPAN VLAN. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.
Step 5	monitor session <i>session_number</i> destination { interface <i>interface-id</i> [, -] [ingress { dot1q vlan <i>vlan-id</i> untagged vlan <i>vlan-id</i> vlan <i>vlan-id</i> }]} Example: <pre>Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6</pre>	Specifies the SPAN session, the destination port, the packet encapsulation, and the incoming VLAN and encapsulation. <ul style="list-style-type: none"> • For <i>session_number</i>, enter the number defined in Step 5. <p>In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port.</p> <ul style="list-style-type: none"> • For <i>interface-id</i>, specify the destination interface. The destination interface must be a physical interface. • Though visible in the command-line help string, encapsulation replicate is not supported for RSPAN.

	Command or Action	Purpose
		<p>The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged.</p> <ul style="list-style-type: none"> • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • Enter ingress with additional keywords to enable forwarding of incoming traffic on the destination port and to specify the encapsulation type: <ul style="list-style-type: none"> • dot1q vlan <i>vlan-id</i>—Forwards incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. • untagged vlan <i>vlan-id</i> or vlan <i>vlan-id</i>—Forwards incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring an FSPAN Session

Follow these steps to create a SPAN session, specify the source (monitored) ports or VLANs and the destination (monitoring) ports, and configure FSPAN for the session.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}

4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [, | -] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation replicate**]}
6. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: <pre>Switch(config)# no monitor session 2</pre>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source { interface <i>interface-id</i> vlan <i>vlan-id</i> } [, -] [both rx tx] Example: <pre>Switch(config)# monitor session 2 source interface gigabitethernet1/0/1</pre>	Specifies the SPAN session and the source port (monitored port). <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>interface-id</i>, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>). Valid port-channel numbers are 1 to 48. • For <i>vlan-id</i>, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). <p>Note A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session.</p>

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) [, -]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) [both rx tx]—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic. <ul style="list-style-type: none"> • both—Monitors both sent and received traffic. This is the default. • rx—Monitors received traffic. • tx—Monitors sent traffic. <p>Note You can use the monitor session session_number source command multiple times to configure multiple source ports.</p>
<p>Step 5</p>	<p>monitor session session_number destination {interface interface-id [, -] [encapsulation replicate]}</p> <p>Example:</p> <pre>Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate</pre>	<p>Specifies the SPAN session and the destination port (monitoring port).</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For destination, specify the following parameters: <ul style="list-style-type: none"> • For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) encapsulation replicate specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). <p>Note For local SPAN, you must use the same session number for the source and destination interfaces.</p> <p>You can use monitor session session_number destination command multiple times to configure multiple destination ports.</p>

	Command or Action	Purpose
Step 6	monitor session <i>session_number</i> filter { ip ipv6 mac } access-group { <i>access-list-number</i> <i>name</i> } Example: <pre>Switch(config)# monitor session 2 filter ipv6 access-group 4</pre>	Specifies the SPAN session, the types of packets to filter, and the ACLs to use in an FSPAN session. <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For <i>access-list-number</i>, specify the ACL number that you want to use to filter traffic. • For <i>name</i>, specify the ACL name that you want to use to filter traffic.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring an FRSPAN Session

Follow these steps to start an RSPAN source session, specify the monitored source and the destination RSPAN VLAN, and configure FRSPAN for the session.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [, | -] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **vlan** *vlan-id*
7. **remote-span**
8. **exit**
9. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
10. **end**
11. **show running-config**

12. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>no monitor session {<i>session_number</i> all local remote}</p> <p>Example:</p> <pre>Switch(config)# no monitor session 2</pre>	<p>Removes any existing SPAN configuration for the session.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	<p>monitor session <i>session_number</i> source {interface <i>interface-id</i> vlan <i>vlan-id</i>} [, -] [both rx tx]</p> <p>Example:</p> <pre>Switch(config)# monitor session 2 source interface gigabitethernet1/0/1</pre>	<p>Specifies the SPAN session and the source port (monitored port).</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>interface-id</i>, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>). Valid port-channel numbers are 1 to 48. • For <i>vlan-id</i>, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). <p>Note A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session.</p> <ul style="list-style-type: none"> • (Optional) [, -]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) [both rx tx]—Specifies the direction of traffic to monitor. If you do not specify a traffic

	Command or Action	Purpose
		<p>direction, the SPAN monitors both sent and received traffic.</p> <ul style="list-style-type: none"> • both—Monitors both sent and received traffic. This is the default. • rx—Monitors received traffic. • tx—Monitors sent traffic. <p>Note You can use the monitor session session_number source command multiple times to configure multiple source ports.</p>
Step 5	<p>monitor session <i>session_number</i> destination remote vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Switch(config)# monitor session 2 destination remote vlan 5</pre>	<p>Specifies the RSPAN session and the destination RSPAN VLAN.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, enter the number defined in Step 4. • For <i>vlan-id</i>, specify the destination RSPAN VLAN to monitor.
Step 6	<p>vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Switch(config)# vlan 10</pre>	<p>Enters the VLAN configuration mode. For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.</p>
Step 7	<p>remote-span</p> <p>Example:</p> <pre>Switch(config-vlan)# remote-span</pre>	<p>Specifies that the VLAN you specified in Step 5 is part of the RSPAN VLAN.</p>
Step 8	<p>exit</p> <p>Example:</p> <pre>Switch(config-vlan)# exit</pre>	<p>Returns to global configuration mode.</p>
Step 9	<p>monitor session <i>session_number</i> filter {<i>ip</i> <i>ipv6</i> <i>mac</i>} access-group {<i>access-list-number</i> <i>name</i>}</p> <p>Example:</p> <pre>Switch(config)# monitor session 2 filter ip access-group 7</pre>	<p>Specifies the RSPAN session, the types of packets to filter, and the ACLs to use in an FRSPAN session.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For <i>access-list-number</i>, specify the ACL number that you want to use to filter traffic. • For <i>name</i>, specify the ACL name that you want to use to filter traffic.

	Command or Action	Purpose
Step 10	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 11	show running-config Example: Switch# show running-config	Verifies your entries.
Step 12	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring SPAN and RSPAN Operations

The following table describes the command used to display SPAN and RSPAN operations configuration and results to monitor operations:

Table 76: Monitoring SPAN and RSPAN Operations

Command	Purpose
show monitor	Displays the current SPAN configuration.

SPAN and RSPAN Configuration Examples

Example: Configuring Local SPAN

This example shows how to set up SPAN session 1 for monitoring source port traffic to a destination port. First, any existing SPAN configuration for session 1 is deleted, and then bidirectional traffic is mirrored from source Gigabit Ethernet port 1 to destination Gigabit Ethernet port 2, retaining the encapsulation method.

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet1/0/1
Switch(config)# monitor session 1 destination interface gigabitethernet1/0/2
encapsulation replicate
Switch(config)# end
```

This example shows how to remove port 1 as a SPAN source for SPAN session 1:

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 1 source interface gigabitethernet1/0/1
Switch(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 1 source interface gigabitethernet1/0/1 rx
```

The monitoring of traffic received on port 1 is disabled, but traffic sent from this port continues to be monitored.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on all ports belonging to VLANs 1 through 3, and send it to destination Gigabit Ethernet port 2. The configuration is then modified to also monitor all traffic on all ports belonging to VLAN 10.

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source vlan 1 - 3 rx
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2
Switch(config)# monitor session 2 source vlan 10
Switch(config)# end
```

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on Gigabit Ethernet source port 1, and send it to destination Gigabit Ethernet port 2 with the same egress encapsulation type as the source port, and to enable ingress forwarding with VLAN 6 as the default ingress VLAN:

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source gigabitethernet0/1 rx
Switch(config)# monitor session 2 destination interface gigabitethernet0/2 encapsulation
replicate ingress vlan 6
Switch(config)# end
```

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor traffic received on Gigabit Ethernet trunk port 2, and send traffic for only VLANs 1 through 5 and VLAN 9 to destination Gigabit Ethernet port 1:

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/1
Switch(config)# end
```

Examples: Creating an RSPAN VLAN

This example shows how to create the RSPAN VLAN 901:

```
Switch> enable
Switch# configure terminal
Switch(config)# vlan 901
Switch(config-vlan)# remote span
Switch(config-vlan)# end
```

This example shows how to remove any existing RSPAN configuration for session 1, configure RSPAN session 1 to monitor multiple source interfaces, and configure the destination as RSPAN VLAN 901:

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet1/0/1 tx
Switch(config)# monitor session 1 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 1 source interface port-channel 2
Switch(config)# monitor session 1 destination remote vlan 901
Switch(config)# end
```

This example shows how to remove any existing configuration on RSPAN session 2, configure RSPAN session 2 to monitor traffic received on trunk port 2, and send traffic for only VLANs 1 through 5 and 9 to destination RSPAN VLAN 902:

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination remote vlan 902
Switch(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN and port 1 as the destination interface:

```
Switch> enable
Switch# configure terminal
Switch(config)# monitor session 1 source remote vlan 901
Switch(config)# monitor session 1 destination interface gigabitethernet2/0/1
Switch(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN in RSPAN session 2, to configure Gigabit Ethernet source port 2 as the destination interface, and to enable forwarding of incoming traffic on the interface with VLAN 6 as the default receiving VLAN:

```
Switch> enable
Switch# configure terminal
Switch(config)# monitor session 2 source remote vlan 901
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6
Switch(config)# end
```




CHAPTER 41

Configuring RMON

- [Finding Feature Information](#), on page 793
- [Information About RMON](#), on page 793
- [How to Configure RMON](#), on page 794
- [Monitoring RMON Status](#), on page 800

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

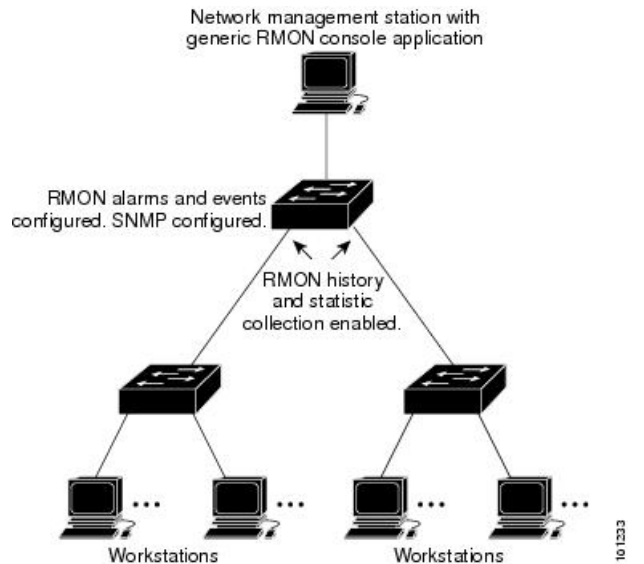
Information About RMON

Understanding RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that defines a set of statistics and functions that can be exchanged between RMON-compliant console systems and network probes. RMON provides comprehensive network-fault diagnosis, planning, and performance-tuning information.

The following figure shows a sample configuration of the RMON feature with the Simple Network Management Protocol (SNMP) agent in the device. This monitors all the traffic flowing among all the devices on all connected LAN segments.

Figure 69: Remote Monitoring Sample



The device supports these RMON groups (defined in RFC 1757):

- Statistics (RMON group 1)—Collects Ethernet statistics (including Fast Ethernet and Gigabit Ethernet statistics, depending on the device type and supported interfaces) on an interface.
- History (RMON group 2)—Collects a history group of statistics on Ethernet ports (including Fast Ethernet and Gigabit Ethernet statistics, depending on the device type and supported interfaces) for a specified polling interval.
- Alarm (RMON group 3)—Monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms can be used with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.
- Event (RMON group 9)—Specifies the action to take when an event is triggered by an alarm. The action can be to generate a log entry or an SNMP trap.

Because devices supported by this software release use hardware counters for RMON data processing, the monitoring is more efficient, and little processing power is required.



Note 64-bit counters are not supported for RMON alarms.

How to Configure RMON

Default RMON Configuration

RMON is disabled by default. No alarms or events are configured.

Configuring RMON Alarms and Events

Before you begin

You can configure your switch for RMON by using the command-line interface (CLI) or an SNMP-compatible network management station.



Note 64-bit counters are not supported for RMON alarms.

Follow these steps to enable RMON alarms and events.

- It is recommended to use a generic RMON console application on the network management station (NMS) to take advantage of the RMON network management capabilities.
- You must also configure SNMP on the switch to access RMON MIB objects.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **rmon alarm** {*number variable interval absolute | delta* } **rising-threshold***value [event-number]* **falling-threshold** *value [event-number]* [**ownerstring**]
4. **rmon event** *number* [**description string**] [**log**] [**owner string**] [**trap community**]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch configure terminal	Enters global configuration mode.
Step 3	rmon alarm { <i>number variable interval absolute delta</i> } rising-threshold <i>value [event-number]</i> falling-threshold <i>value [event-number]</i> [ownerstring] Example: Switch(config)# rmon alarm 10 ifEntry.20.1 20 delta rising-threshold 15 1 falling-threshold 0 owner jjohnson	Sets an alarm on a MIB object. For <i>number</i> , specify the alarm number. The range is 1 to 65535. For <i>variable</i> , specify the MIB object to monitor For <i>interval</i> , specify the time in seconds the alarm monitors the MIB variable. The range is 1 to 4294967295 seconds.

	Command or Action	Purpose
		<p>Specify the absolute keyword to test each MIB variable directly. Specify the delta keyword to test the change between samples of a MIB variable.</p> <p>For <i>value</i>, specify a number at which the alarm is triggered and one for when the alarm is reset. The range for the rising threshold and falling threshold values is -2147483648 to 2147483647.</p> <p>(Optional) For <i>event-number</i>, specify the event number to trigger when the rising or falling threshold exceeds its limit.</p> <p>(Optional) For owner string, specify the owner of the alarm.</p>
Step 4	<p>rmon event <i>number</i> [description string] [log] [owner string] [trap community]</p> <p>Example:</p> <pre>Switch(config)# rmon event 1 log trap eventtrap description "High ifOutErrors" owner jjones</pre>	<p>Adds an event in the RMON event table that is associated with an RMON event number.</p> <p>For <i>number</i>, assign an event number. The range is 1 to 65535.</p> <p>(Optional) For description string, specify a description of the event.</p> <p>(Optional) Use the log keyword to generate an RMON log entry when the event is triggered.</p> <p>(Optional) For owner string, specify the owner of this event.</p> <p>(Optional) For trap community, enter the SNMP community string used for this trap.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To disable an alarm, use the **no rmon alarm** *number* global configuration command on each alarm you configured. You cannot disable at once all the alarms that you configured. To disable an event, use the **no rmon event** *number* global configuration command.

Collecting Group History Statistics on an Interface

Follow these steps to collect group history statistics on an interface. This procedure is optional.

Before you begin

You must first configure RMON alarms and events to display collection information.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **rmon collection history** *index* [**buckets** *bucket-number*] [**interval** *seconds*] [**owner** *ownername*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/1	Specifies the interface on which to collect history, and enter interface configuration mode.
Step 4	rmon collection history <i>index</i> [buckets <i>bucket-number</i>] [interval <i>seconds</i>] [owner <i>ownername</i>] Example:	Enables history collection for the specified number of buckets and time period. For <i>index</i> , identify the RMON group of statistics. The range is 1 to 65535. (Optional) For buckets <i>bucket-number</i> , specify the maximum number of buckets desired for the RMON

	Command or Action	Purpose
		<p>collection history group of statistics. The range is 1 to 65535. The default is 50 buckets.</p> <p>(Optional) For interval <i>seconds</i>, specify the number of seconds in each polling cycle. The range is 1 to 3600. The default is 1800 seconds.</p> <p>(Optional) For owner <i>ownername</i>, enter the name of the owner of the RMON group of statistics.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To disable history collection, use the **no rmon collection history** *index* interface configuration command.

Collecting Group Ethernet Statistics on an Interface

Follow these steps to collect group Ethernet statistics on an interface. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **rmon collection stats** *index* [**owner** *ownername*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/1	Specifies the interface on which to collect statistics, and enter interface configuration mode.
Step 4	rmon collection stats <i>index</i> [owner <i>ownername</i>] Example: Switch(config-if)# rmon collection stats 2 owner root	Enables RMON statistic collection on the interface. For <i>index</i> , specify the RMON group of statistics. The range is from 1 to 65535. (Optional) For owner <i>ownername</i> , enter the name of the owner of the RMON group of statistics.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

To disable the collection of group Ethernet statistics, use the **no rmon collection stats index** interface configuration command.

Monitoring RMON Status

Table 77: Commands for Displaying RMON Status

Command	Purpose
<code>show rmon</code>	Displays general RMON statistics.
<code>show rmon alarms</code>	Displays the RMON alarm table.
<code>show rmon events</code>	Displays the RMON event table.
<code>show rmon history</code>	Displays the RMON history table.
<code>show rmon statistics</code>	Displays the RMON statistics table.



CHAPTER 42

Configuring Embedded Event Manager

- [Information about Embedded Event Manager, on page 801](#)
- [How to Configure Embedded Event Manager, on page 804](#)
- [Monitoring Embedded Event Manager, on page 806](#)
- [Configuration Examples for Embedded Event Manager, on page 807](#)

Information about Embedded Event Manager

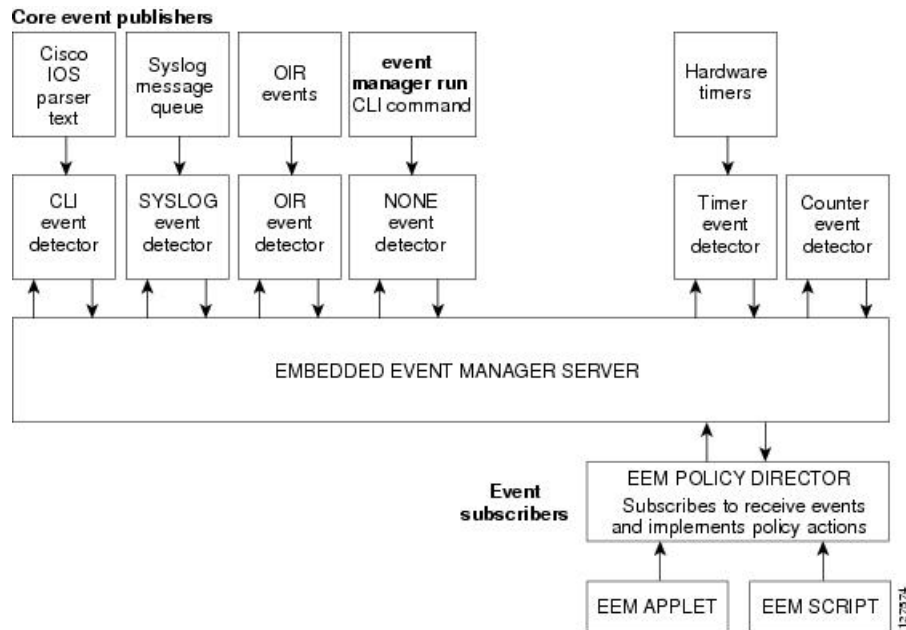
Understanding Embedded Event Manager

Embedded Event Manager (EEM) is a distributed and customized approach to event detection and recovery within a Cisco IOS device. EEM offers the ability to monitor events and take informational, corrective, or any other EEM action when the monitored events occur or when a threshold is reached. An EEM policy defines an event and the actions to be taken when that event occurs.

EEM monitors key system events and then acts on them through a set policy. This policy is a programmed script that you can use to customize a script to invoke an action based on a given set of events occurring. The script generates actions such as generating custom syslog or Simple Network Management Protocol (SNMP) traps, invoking CLI commands, forcing a failover, and so forth. The event management capabilities of EEM are useful because not all event management can be managed from the switch and because some problems compromise communication between the switch and the external network management device. Network availability is improved if automatic recovery actions are performed without rebooting the switch.

This example shows the relationship between the EEM server, the core event publishers (event detectors), and the event subscribers (policies). The event publishers screen events and when there is a match on an event specification that is provided by the event subscriber. Event detectors notify the EEM server when an event occurs. The EEM policies then implement recovery based on the current state of the system and the actions specified in the policy for the given event.

Figure 70: Embedded Event Manager Core Event Detectors



- Note** EEM is supported only on Cisco Catalyst 3560-CX switches.
EEM is supported only on Catalyst switches running IP Base and IP Services licenses.

Embedded Event Manager Actions

These actions occur in response to an event:

- Modifying a named counter.
- Publishing an application-specific event.
- Generating an SNMP trap.
- Generating prioritized syslog messages.
- Reloading the Cisco IOS software.
- Reloading the switch stack.
- Reloading the master switch in the event of a master switchover. If this occurs, a new master switch is elected.

Embedded Event Manager Policies

EEM can monitor events and provide information, or take corrective action when the monitored events occur or a threshold is reached. An EEM policy is an entity that defines an event and the actions to be taken when that event occurs.

There are two types of EEM policies: an applet or a script. An applet is a simple policy that is defined within the CLI configuration. It is a concise method for defining event screening criteria and the actions to be taken

when that event occurs. Scripts are defined on the networking device by using an ASCII editor. The script, which can be a bytecode (.tbc) and text (.tcl) script, is then copied to the networking device and registered with EEM. You can also register multiple events in a .tcl file.

You use EEM to write and implement your own policies using the EEM policy tool command language (TCL) script. When you configure a TCL script on the master switch and the file is automatically sent to the member switches. The user-defined TCL scripts must be available in the member switches so that if the master switch changes, the TCL scripts policies continue to work.

Cisco enhancements to TCL in the form of keyword extensions facilitate the development of EEM policies. These keywords identify the detected event, the subsequent action, utility information, counter values, and system information.

Embedded Event Manager Environment Variables

EEM uses environment variables in EEM policies. These variables are defined in a EEM policy tool command language (TCL) script by running a CLI command and the **event manager environment** command.

- User-defined variables —Defined by the user for a user-defined policy.
- Cisco-defined variables —Defined by Cisco for a specific sample policy.
- Cisco built-in variables (available in EEM applets) —Defined by Cisco and can be read-only or read-write. The read-only variables are set by the system before an applet starts to execute. The single read-write variable, `_exit_status`, allows you to set the exit status for policies triggered from synchronous events.

Cisco-defined environment variables and Cisco system-defined environment variables might apply to one specific event detector or to all event detectors. Environment variables that are user-defined or defined by Cisco in a sample policy are set by using the event manager environment global configuration command. You must defined the variables in the EEM policy before you register the policy.

Embedded Event Manager 3.2

Embedded Event Manager 3.2 provides support for the following event detectors:

- Neighbor Discovery—Neighbor Discovery event detector provides the ability to publish a policy to respond to automatic neighbor detection when:
 - a Cisco Discovery Protocol (CDP) cache entry is added, deleted, or updated.
 - a Link Layer Discovery Protocol (LLDP) cache entry is added, deleted or updated.
 - an interface link status changes.
 - an interface line status changes.
- Identity—Identity event detector generates an event when AAA authorization and authentication is successful, when failure occurs, or after normal user traffic on the port is allowed to flow.
- Mac-Address-Table—Mac-Address-Table event detector generates an event when a MAC address is learned in the MAC address table.



Note The Mac-Address-Table event detector is supported only on switch platforms and can be used only on Layer 2 interfaces where MAC addresses are learned. Layer 3 interfaces do not learn addresses, and routers do not usually support the MAC address-table infrastructure needed to notify EEM of a learned MAC address.

EEM 3.2 also introduces CLI commands to support the applets to work with the new event detectors.

How to Configure Embedded Event Manager

Registering and Defining an Embedded Event Manager Applet

Beginning in privileged EXEC mode, perform this task to register an applet with EEM and to define the EEM applet using the **event applet** and **action applet** configuration commands.



Note Only one event applet command is allowed in an EEM applet. Multiple action applet commands are permitted. If you do not specify the **no event** and **no action** commands, the applet is removed when you exit configuration mode.

SUMMARY STEPS

1. **configure terminal**
2. **event manager applet***applet-name*
3. **event snmp oid** *oid-value* **get-type** {**exact**|**next**} **entry-op** { **eq**|**ge**|**gt**|**le**|**lt**|**ne**} **entry-val** *entry-val* [**exit-comb** {**or**|**and**}] [**exit-op** {**eq**|**ge**|**gt**|**le**|**lt**|**nc**}] [**exit-val** *exit-val*] [**exit-time** *exit-time-val*] **poll interval** *poll-int-val*
4. **action label** **syslog** [**priority** *priority-level*] **msg** *msg-text*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	event manager applet <i>applet-name</i>	Register the applet with EEM and enter applet configuration mode.
Step 3	event snmp oid <i>oid-value</i> get-type { exact next } entry-op { eq ge gt le lt ne } entry-val <i>entry-val</i> [exit-comb { or and }] [exit-op { eq ge gt le lt nc }] [exit-val <i>exit-val</i>] [exit-time <i>exit-time-val</i>] poll interval <i>poll-int-val</i>	Specify the event criteria that causes the EEM applet to run. (Optional) Exit criteria. If exit criteria are not specified, event monitoring is re-enabled immediately.
Step 4	action label syslog [priority <i>priority-level</i>] msg <i>msg-text</i>	Specify the action when an EEM applet is triggered. Repeat this action to add other CLI commands to the applet. <ul style="list-style-type: none"> • (Optional) The priority keyword specifies the priority level of the syslog messages. If selected, you need to define the priority-level argument. • For <i>msg-text</i>, the argument can be character text, an environment variable, or a combination of the two.

	Command or Action	Purpose
Step 5	end	Exit applet configuration mode and return to privileged EXEC mode.

Example

This example shows the output for EEM when one of the fields specified by an SNMP object ID crosses a defined threshold:

```
Switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
lt entry-val 512000 poll-interval 10
```

These examples show actions that are taken in response to an EEM event:

```
Switch(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current
available memory is $_snmp_oid_val bytes"
Switch (config-applet)# action 2.0 force-switchover
```

Registering and Defining an Embedded Event Manager TCL Script

Beginning in privileged EXEC mode, perform this task to register a TCL script with EEM and to define the TCL script and policy commands.

SUMMARY STEPS

1. configure terminal
2. show event manager environment [all | variable-name]
3. configure terminal
4. event manager environment variable-name string
5. event manager policy policy-file-name [type system] [trap]
6. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal	Enters the global configuration mode.
Step 2	show event manager environment [all variable-name]	(Optional) The show event manager environment command displays the name and value of the EEM environment variables. (Optional) The all keyword displays the EEM environment variables. (Optional) The <i>variable-name</i> argument displays information about the specified environment variable.
Step 3	configure terminal	Enters the global configuration mode.

	Command or Action	Purpose
Step 4	<code>event manager environment variable-name string</code>	Configures the value of the specified EEM environment variable. Repeat this step for all the required environment variables.
Step 5	<code>event manager policy policy-file-name [type system] [trap]</code>	Registers the EEM policy to be run when the specified event defined within the policy occurs.
Step 6	<code>exit</code>	Exits the global configuration mode and return to the privileged EXEC mode.

Example

This example shows the sample output for the show event manager environment command:

```
Switch# show event manager environment all
No.   Name                Value
1     _cron_entry          0-59/2 0-23/1 * * 0-6
2     _show_cmd            show ver
3     _syslog_pattern      .*UPDOWN.*Ethernet1/0.*
```

This example shows a CRON timer environment variable, which is assigned by the software, to be set to every second minute, every hour of every day:

```
Switch (config)# event manager environment_cron_entry 0-59/2 0-23/1 * * 0-6
```

This example shows the sample EEM policy named tm_cli_cmd.tcl registered as a system policy. The system policies are part of the Cisco IOS image. User-defined TCL scripts must first be copied to flash memory.

```
Switch (config)# event manager policy tm_cli_cmd.tcl type system
```

Monitoring Embedded Event Manager

Displaying Embedded Event Manager Information

Table 78: Commands for displaying EEM information

Command	Purpose
<code>show event manager environment[all variable-name]</code>	Displays the name and value of the EEM environment variables.

To display information about EEM, including EEM registered policies and EEM history data, see the [Cisco IOS Network Management Command Reference](#).

Configuration Examples for Embedded Event Manager

Example: Generating SNMP Notifications

This example shows the output for EEM when one of the fields specified by an SNMP object ID crosses a defined threshold.

```
Switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
lt entry-val 5120000 poll-interval 10
```

Example: Responding to EEM Events

These examples show actions that are taken in response to an EEM event:

```
Switch(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current
available memory is $_snmp_oid_val bytes"
Switch(config-applet)# action 2.0 force-switchover
```

Example: Displaying EEM Environment Variables

This example shows the sample output for the show event manager environment command:

```
Switch# show event manager environment all
No.   Name                               Value
1     _cron_entry                          0-59/2 0-23/1 * * 0-6
2     _show_cmd                            show ver
3     _syslog_pattern                      .*UPDOWN.*Ethernet1/0.*
4     _config_cmd1 interface              Ethernet1/0
5     _config_cmd2                         no shut
```

This example shows a CRON timer environment variable, which is assigned by the software, to be set to every second minute, every hour of every day:

```
Switch(config)# event manager environment_cron_entry 0-59/2 0-23/1 * * 0-6
```

This example shows the sample EEM policy named tm_cli_cmd.tcl registered as a system policy. The system policies are part of the Cisco IOS image. User-defined TCL scripts must first be copied to flash memory.

```
Switch(config)# event manager policy tm_cli_cmd.tcl type system
```




CHAPTER 43

Configuring Flexible NetFlow

- [Finding Feature Information, on page 809](#)
- [Prerequisites for NetFlow Lite, on page 809](#)
- [Restrictions for NetFlow Lite, on page 809](#)
- [Information About NetFlow Lite, on page 811](#)
- [How to Configure Flexible Netflow, on page 818](#)
- [Monitoring Flexible NetFlow, on page 829](#)
- [Configuration Examples for NetFlow Lite, on page 830](#)
- [Feature Information for Flexible NetFlow, on page 831](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for NetFlow Lite

The following two targets for attaching a NetFlow Lite monitor are supported:

- Port—Monitor attachment is only supported on physical interfaces and not on logical interfaces, such as EtherChannels. The physical interface could be a routed port or a switched port.
- VLAN—Monitor attachment is supported on VLAN interfaces only (SVI) and not on a Layer 2 VLAN.

Restrictions for NetFlow Lite

The following are restrictions for NetFlow Lite:

- Monitor restrictions:

- Monitor attachment is only supported in the ingress direction.
 - One monitor per interface is supported, although multiple exporters per interface are supported.
 - Only permanent and normal cache is supported for the monitor; immediate cache is not supported.
 - Changing any monitor parameter will not be supported when it is applied on any of the interfaces or VLANs.
 - When both the port and VLANs have monitors attached, then VLAN monitor will overwrite the port monitor for traffic coming on the port.
 - Flow monitor type and traffic type (type means IPv4, IPv6, and data link) should be same for the flows to be created.
 - You cannot attach an IP and port-based monitor to an interface at the same time on the device. A 48-port device supports a maximum of 48 monitors (IP or port-based) and for 256 SVIs, you can configure up to 256 monitors (IP or port-based).
 - When running the **show flow monitor *flow_name* cache** command, the switch displays cache information from an earlier switch software version (Catalyst 2960-S) with all fields entered as zero. Ignore these fields, as they are inapplicable to the switch.
- Sampler restrictions:
 - Only sampled NetFlow is supported.
 - For both port and VLANS, a total of only 4 samplers (random or deterministic) are supported on the device.
 - The sampling minimum rate for both modes is 1 out of 32 flows, and the sampling maximum rate for both modes is 1 out of 1022 flows.
 - You must associate a sampler with a monitor while attaching it to an interface. Otherwise, the command will be rejected. Use the **ip flow monitor *monitor_name* sampler *sampler_name* input** interface configuration command to perform this task.
 - When you attach a monitor using a deterministic sampler, every attachment with the same sampler uses one new free sampler from the switch (hardware) out of 4 available samplers. You are not allowed to attach a monitor with any sampler, beyond 4 attachments.

When you attach a monitor using a random sampler, only the first attachment uses a new sampler from the switch (hardware). The remainder of all of the attachments using the same sampler, share the same sampler.

Because of this behavior, when using a deterministic sampler, you can always make sure that the correct number of flows are sampled by comparing the sampling rate and what the device sends. If the same random sampler is used with multiple interfaces, flows from any interface can always be sampled, and flows from other interfaces can always be skipped.
 - Network flows and statistics are collected at the line rate.
 - ACL-based NetFlow is not supported.
 - Only NetFlow Version 9 is supported for Flexible NetFlow exporter using the *export-protocol* command option. If you configure NetFlow Version 5, this version will be accepted, but the NetFlow Version 5 export functionality is neither currently available nor supported.

- The switch supports homogeneous stacking, but does not support mixed stacking.

Information About NetFlow Lite

NetFlow Lite Overview

NetFlow Lite uses flows to provide statistics for accounting, network monitoring, and network planning.

A flow is a unidirectional stream of packets that arrives on a source interface and has the same values for the keys. A key is an identified value for a field within the packet. You create a flow using a flow record to define the unique keys for your flow.

The device supports the NetFlow Lite feature that enables enhanced network anomalies and security detection. NetFlow Lite allows you to define an optimal flow record for a particular application by selecting the keys from a large collection of predefined fields.

All key values must match for the packet to count in a given flow. A flow might gather other fields of interest, depending on the export record version that you configure. Flows are stored in the NetFlow Lite cache.

You can export the data that NetFlow Lite gathers for your flow by using an exporter and export this data to a remote system such as a NetFlow Lite collector. The NetFlow Lite collector can use an IPv4 address.

You define the size of the data that you want to collect for a flow using a monitor. The monitor combines the flow record and exporter with the NetFlow Lite cache information.

Starting with the Cisco IOS XE 16.12.1 release, Source Group Tag (SGT) and Destination Group Tag (DGT) fields over Flexible NetFlow are supported for IPv6 traffic.

Flexible NetFlow Components

Flexible NetFlow consists of components that can be used together in several variations to perform traffic analysis and data export. The user-defined flow records and the component structure of Flexible NetFlow facilitates the creation of various configurations for traffic analysis and data export on a networking device with a minimum number of configuration commands. Each flow monitor can have a unique combination of flow record, flow exporter, and cache type. If you change a parameter such as the destination IP address for a flow exporter, it is automatically changed for all the flow monitors that use the flow exporter. The same flow monitor can be used in conjunction with different flow samplers to sample the same type of network traffic at different rates on different interfaces. The following sections provide more information on Flexible NetFlow components:

Flow Records

In Flexible NetFlow a combination of key and nonkey fields is called a record. Flexible NetFlow records are assigned to Flexible NetFlow flow monitors to define the cache that is used for storing flow data.

A flow record defines the keys that Flexible NetFlow uses to identify packets in the flow, as well as other fields of interest that Flexible NetFlow gathers for the flow. You can define a flow record with any combination of keys and fields of interest. The device supports a rich set of keys. A flow record also defines the types of counters gathered per flow. You can configure 64-bit packet or byte counters. The device enables the following match fields as the defaults when you create a flow record:

- match datalink—Layer 2 attributes

- match ipv4—IPv4 attributes
- match ipv6—IPv6 attributes
- match transport—Transport layer fields
- match wireless—Wireless fields

NetFlow Predefined Records

Flexible NetFlow includes several predefined records that you can use to start monitoring traffic in your network. The predefined records are available to help you quickly deploy Flexible NetFlow and are easier to use than user-defined flow records. You can choose from a list of already defined records that may meet the needs for network monitoring. As Flexible NetFlow evolves, popular user-defined flow records will be made available as predefined records to make them easier to implement.

The predefined records ensure backward compatibility with your existing NetFlow collector configurations for the data that is exported. Each of the predefined records has a unique combination of key and nonkey fields that offer you the built-in ability to monitor various types of traffic in your network without customizing Flexible NetFlow on your router.

Two of the predefined records (NetFlow original and NetFlow IPv4/IPv6 original output), which are functionally equivalent, emulate original (ingress) NetFlow and the Egress NetFlow Accounting feature in original NetFlow, respectively. Some of the other Flexible NetFlow predefined records are based on the aggregation cache schemes available in original NetFlow. The Flexible NetFlow predefined records that are based on the aggregation cache schemes available in original NetFlow do not perform aggregation. Instead each flow is tracked separately by the predefined records.

User-Defined Records

Flexible NetFlow enables you to define your own records for a Flexible NetFlow flow monitor cache by specifying the key and nonkey fields to customize the data collection to your specific requirements. When you define your own records for a Flexible NetFlow flow monitor cache, they are referred to as *user-defined records*. The values in nonkey fields are added to flows to provide additional information about the traffic in the flows. A change in the value of a nonkey field does not create a new flow. In most cases the values for nonkey fields are taken from only the first packet in the flow. Flexible NetFlow enables you to capture counter values such as the number of bytes and packets in a flow as nonkey fields.

Flexible NetFlow adds a new Version 9 export format field type for the header and packet section types. Flexible NetFlow will communicate to the NetFlow collector the configured section sizes in the corresponding Version 9 export template fields. The payload sections will have a corresponding length field that can be used to collect the actual size of the collected section.

NetFlow Lite Match Parameters

You can match these key fields for the flow record:

- IPv4 or IPv6 destination address
- Datalink fields (source and destination MAC address, and MAC ethertype (type of networking protocol)).
- Transport field source and destination ports to identify the type of application: ICMP, IGMP, or TCP traffic.

The following table describes NetFlow Lite match parameters. You must configure at least one of the following match parameters for the flow records.

Table 79: Match Parameters

Command	Purpose
match datalink { ethertype mac { destination address input source address input }}	<p>Specifies a match to datalink or Layer 2 fields. The following command options are available:</p> <ul style="list-style-type: none"> • ethertype—Matches to the ethertype of the packet. • mac—Matches the source or destination MAC address from packets at input. <p>Note When a datalink flow monitor is assigned to an interface or VLAN, it only creates flows for non-IPv6 or non-IPv4 traffic.</p>
match ipv4 { destination { address } protocol source { address } tos }	<p>Specifies a match to the IPv4 fields. The following command options are available:</p> <ul style="list-style-type: none"> • destination—Matches to the IPv4 destination address-based fields. • protocol—Matches to the IPv4 protocols. • source—Matches to the IPv4 source address based fields. • tos—Matches to the IPv4 Type of Service fields.
match ipv6 { destination { address } flow-label protocol source { address } }	<p>Specifies a match to the IPv6 fields. The following command options are available:</p> <ul style="list-style-type: none"> • destination—Matches to the IPv6 destination address-based fields. • flow-label—Matches to the IPv6 flow-label fields. • protocol—Matches to the IPv6 payload protocol fields. • source—Matches to the IPv6 source address based fields.
match transport { destination-port source-port }	<p>Specifies a match to the Transport Layer fields. The following command options are available:</p> <ul style="list-style-type: none"> • destination-port—Matches to the transport destination port. • source-port—Matches to the transport source port.

Command	Purpose
	Specifies the use of SSID of the wireless network as a key field for a flow record.

NetFlow Lite Collect Parameters

You can collect these key fields in the flow record:

- The total number of bytes, flows or packets sent by the exporter (exporter) or the number of bytes or packets in a 64-bit counter (long).
- The timestamp based on system uptime from the time the first packet was sent or from the time the most recent (last) packet was seen.
- The SNMP index of the input interface. The interface for traffic entering the service module is based on the switch forwarding cache. This field is typically used in conjunction with datalink, IPv4, and IPv6 addresses, and provides the actual first-hop interface for directly connected hosts.
 - A value of 0 means that interface information is not available in the cache.
 - Some NetFlow collectors require this information in the flow record.

The following table describes NetFlow Lite collect parameters.

Table 80: Collect Parameters

Command	Purpose
collect counter {bytes {long permanent} packets { long permanent}}	Collects the counter fields total bytes and total packets.
collect flow {sampler}	Collects the flow sampler identifier (ID).
collect interface {input}	Collects the fields from the input interface.
collect timestamp sys-uptime {first last}	Collects the fields for the time the first packet was seen or the time the most recent packet was last seen (in milliseconds).
collect transport tcp flags	Collects the following transport TCP flags: <ul style="list-style-type: none"> • ack—TCP acknowledgement flag • cwr—TCP congestion window reduced flag • ece—TCP ECN echo flag • fin—TCP finish flag • psh—TCP push flag • rst—TCP reset flag • syn—TCP synchronize flag • urg—TCP urgent flag

Command	Purpose
	Collects the MAC addresses of the access points that the wireless client is associated with.

Flow Exporters

Flow exporters export the data in the flow monitor cache to a remote system, such as a server running NetFlow collector, for analysis and storage. Flow exporters are created as separate entities in the configuration. Flow exporters are assigned to flow monitors to provide data export capability for the flow monitors. You can create several flow exporters and assign them to one or more flow monitors to provide several export destinations. You can create one flow exporter and apply it to several flow monitors.

NetFlow Data Export Format Version 9

The basic output of NetFlow is a flow record. Several different formats for flow records have evolved as NetFlow has matured. The most recent evolution of the NetFlow export format is known as Version 9. The distinguishing feature of the NetFlow Version 9 export format is that it is template-based. Templates provide an extensible design to the record format, a feature that should allow future enhancements to NetFlow services without requiring concurrent changes to the basic flow-record format. Using templates provides several key benefits:

- Third-party business partners who produce applications that provide collector or display services for NetFlow do not have to recompile their applications each time a new NetFlow feature is added. Instead, they should be able to use an external data file that documents the known template formats.
- New features can be added to NetFlow quickly without breaking current implementations.
- NetFlow is “future-proofed” against new or developing protocols because the Version 9 format can be adapted to provide support for them.

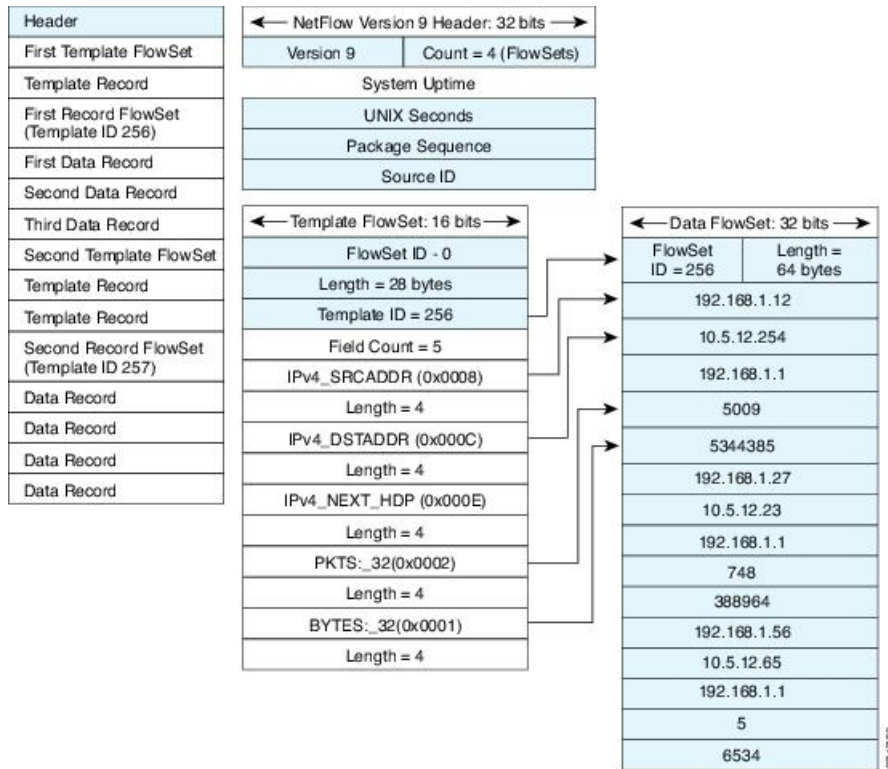
The Version 9 export format consists of a packet header followed by one or more template flow or data flow sets. A template flow set provides a description of the fields that will be present in future data flow sets. These data flow sets may occur later within the same export packet or in subsequent export packets. Template flow and data flow sets can be intermingled within a single export packet, as illustrated in the figure below.

Figure 71: Version 9 Export Packet



NetFlow Version 9 will periodically export the template data so the NetFlow collector will understand what data is to be sent and also export the data flow set for the template. The key advantage to Flexible NetFlow is that the user configures a flow record, which is effectively converted to a Version 9 template and then forwarded to the collector. The figure below is a detailed example of the NetFlow Version 9 export format, including the header, template flow, and data flow sets.

Figure 72: Detailed Example of the NetFlow Version 9 Export Format



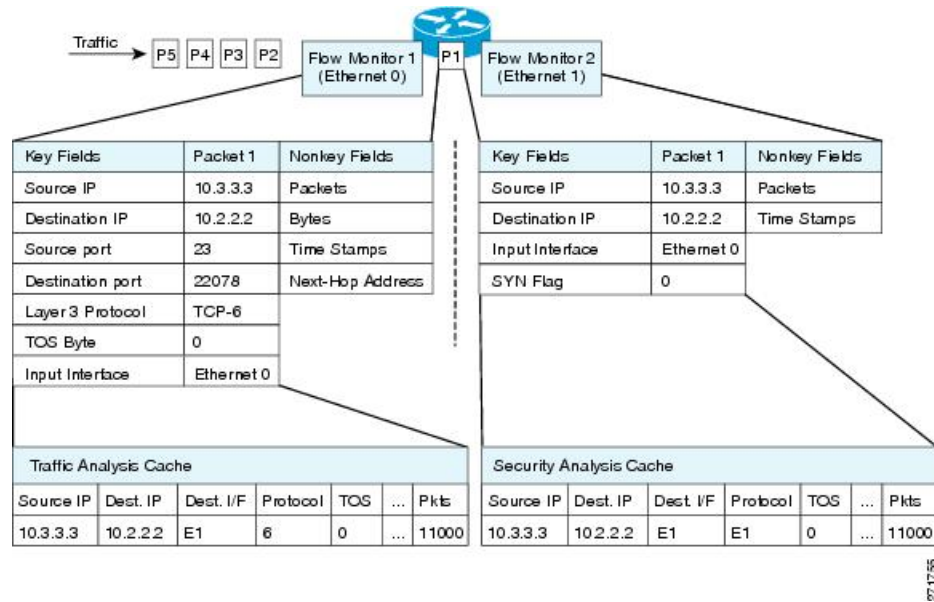
Flow Monitors

Flow monitors are the Flexible NetFlow component that is applied to interfaces to perform network traffic monitoring.

Flow data is collected from the network traffic and added to the flow monitor cache during the monitoring process based on the key and nonkey fields in the flow record.

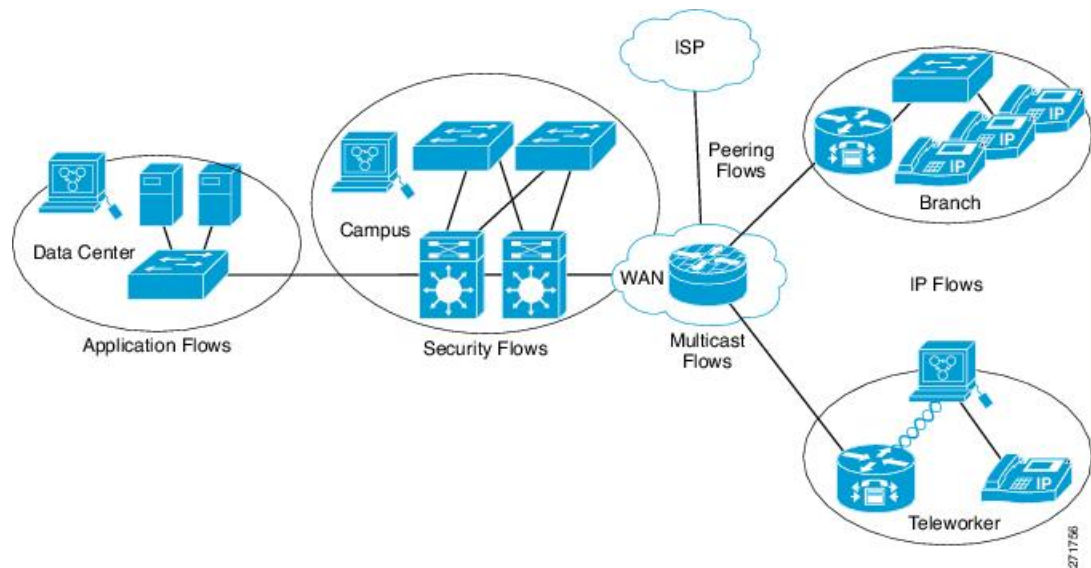
Flexible NetFlow can be used to perform different types of analysis on the same traffic. In the figure below, packet 1 is analyzed using a record designed for standard traffic analysis on the input interface and a record designed for security analysis on the output interface.

Figure 73: Example of Using Two Flow Monitors to Analyze the Same Traffic



The figure below shows a more complex example of how you can apply different types of flow monitors with custom records.

Figure 74: Complex Example of Using Multiple Types of Flow Monitors with Custom Records



Normal

The default cache type is “normal”. In this mode, the entries in the cache are aged out according to the timeout active and timeout inactive settings. When a cache entry is aged out, it is removed from the cache and exported via any exporters configured.

Flow Samplers

Flow samplers are created as separate components in a router's configuration. Flow samplers are used to reduce the load on the device that is running NetFlow Lite by limiting the number of packets that are selected for analysis.

Flow sampling exchanges monitoring accuracy for router performance. When you apply a sampler to a flow monitor, the overhead load on the router of running the flow monitor is reduced because the number of packets that the flow monitor must analyze is reduced. The reduction in the number of packets that are analyzed by the flow monitor causes a corresponding reduction in the accuracy of the information stored in the flow monitor's cache.

Samplers are combined with flow monitors when they are applied to an interface with the **ip flow monitor** command.

Default Settings

The following table lists the NetFlow Lite default settings for the device.

Table 81: Default NetFlow Lite Settings

Setting	Default
Flow active timeout	1800 seconds Note The default value for this setting may be too high for your specific NetFlow Lite configuration. You may want to consider changing it to a lower value of 180 or 300 seconds.
Flow timeout inactive	Enabled, 30 seconds
Flow update timeout	1800 seconds
Default cache size	16640 entries

How to Configure Flexible Netflow

To configure Flexible Netflow, follow these general steps:

1. Create a flow record by specifying keys and non-key fields to the flow.
2. Create an optional flow exporter by specifying the protocol and transport destination port, destination, and other parameters.
3. Create a flow monitor based on the flow record and flow exporter.
4. Create an optional sampler.
5. Apply the flow monitor to a Layer 2 port, Layer 3 port, or VLAN.

Creating a Flow Record

You can create a flow record and add keys to match on and fields to collect in the flow.

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *name*
3. **description** *string*
4. **match** *type*
5. **collect** *type*
6. **end**
7. **show flow record** [**name** *record-name*]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	flow record <i>name</i> Example: Switch(config)# flow record test Switch(config-flow-record)#	Creates a flow record and enters flow record configuration mode.
Step 3	description <i>string</i> Example: Switch(config-flow-record)# description Ipv4Flow	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	match <i>type</i> Example: Switch(config-flow-record)# match ipv4 source address Switch(config-flow-record)# match ipv4 destination address Switch(config-flow-record)# match flow direction	Specifies a match key.
Step 5	collect <i>type</i> Example:	Specifies the collection field.

	Command or Action	Purpose
	<pre>Switch(config-flow-record)# collect counter bytes layer2 long Switch(config-flow-record)# collect counter bytes long Switch(config-flow-record)# collect timestamp absolute first Switch(config-flow-record)# collect transport tcp flags Switch(config-flow-record)# collect interface output</pre>	<p>Note</p> <p>When a flow monitor has the collect interface output as the collect field in the flow record, then the output interface is detected based on the destination address in the switch. Hence, for the different flow monitors, the following are required to be configured:</p> <ul style="list-style-type: none"> • For ipv4 flow monitor, configure "match ip destination address" • For ipv6 flow monitor, configure "match ipv6 destination address" • For datalink flow monitor, configure "match datalink mac output" <p>The collect interface output field will return a value of NULL when a flow gets created for any of the following addresses:</p> <ul style="list-style-type: none"> • L3 broadcast • L2 broadcast • L3 Multicast • L2 Multicast • L2 unknown destination.
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config-flow-record)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show flow record [name record-name]</p> <p>Example:</p> <pre>Switch show flow record test</pre>	(Optional) Displays information about NetFlow flow records.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Define an optional flow exporter by specifying the export format, protocol, destination, and other parameters.

Creating a Flow Exporter

You can create a flow export to define the export parameters for a flow.



Note Each flow exporter supports only one destination. If you want to export the data to multiple destinations, you must configure multiple flow exporters and assign them to the flow monitor.

You can export to a destination using IPv4 address.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** *name*
3. **description** *string*
4. **destination** {*ipv4-address*}
5. **dscp** *value*
6. **transport udp** *number*
7. **ttl** *seconds*
8. **export-protocol** {*netflow-v9*}
9. **end**
10. **show flow exporter** [*name record-name*]
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	flow exporter <i>name</i> Example: Switch(config)# flow exporter ExportTest	Creates a flow exporter and enters flow exporter configuration mode.
Step 3	description <i>string</i> Example: Switch(config-flow-exporter)# description ExportV9	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	destination { <i>ipv4-address</i> } Example:	Sets the IPv4 destination address or hostname for this exporter.

	Command or Action	Purpose
	Switch(config-flow-exporter) # destination 192.0.2.1 (IPv4 destination)	
Step 5	dscp value Example: Switch(config-flow-exporter) # dscp 0	(Optional) Specifies the differentiated services codepoint value. The range is from 0 to 63. The default is 0.
Step 6	transport udp number Example: Switch(config-flow-exporter) # transport udp 200	(Optional) Specifies the UDP port to use to reach the NetFlow collector.
Step 7	ttl seconds Example: Switch(config-flow-exporter) # ttl 210	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. The range is from 1 to 255 seconds. The default is 255.
Step 8	export-protocol {netflow-v9} Example: Switch(config-flow-exporter) # export-protocol netflow-v9	Specifies the version of the NetFlow export protocol used by the exporter.
Step 9	end Example: Switch(config-flow-record) # end	Returns to privileged EXEC mode.
Step 10	show flow exporter [name record-name] Example: Switch# show flow exporter ExportTest	(Optional) Displays information about NetFlow flow exporters.
Step 11	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Define a flow monitor based on the flow record and flow exporter.

Creating a Flow Monitor

You can create a flow monitor and associate it with a flow record and a flow exporter.

SUMMARY STEPS

1. **configure terminal**
2. **flow monitor** *name*
3. **description** *string*
4. **exporter** *name*
5. **record** *name*
6. **cache** { **timeout** { **active** | **inactive** } *seconds* | **type normal** }
7. **end**
8. **show flow monitor** [**name** *record-name*]
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	flow monitor <i>name</i> Example: <pre>Switch(config)# flow monitor MonitorTest Switch (config-flow-monitor)#</pre>	Creates a flow monitor and enters flow monitor configuration mode.
Step 3	description <i>string</i> Example: <pre>Switch(config-flow-monitor)# description Ipv4Monitor</pre>	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	exporter <i>name</i> Example: <pre>Switch(config-flow-monitor)# exporter ExportTest</pre>	Associates a flow exporter with this flow monitor.
Step 5	record <i>name</i>	Associates a flow record with the specified flow monitor.

	Command or Action	Purpose
	Example: Switch(config-flow-monitor)# record test	
Step 6	cache { timeout {active inactive} seconds type normal } Example: Switch(config-flow-monitor)# cache timeout active 15000	Associates a flow cache with the specified flow monitor.
Step 7	end Example: Switch(config-flow-monitor)# end	Returns to privileged EXEC mode.
Step 8	show flow monitor [name record-name] Example: Switch show flow monitor name MonitorTest	(Optional) Displays information about NetFlow flow monitors.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Apply the flow monitor to a Layer 2 interface, Layer 3 interface, or VLAN.

Creating a Sampler

You can create a sampler to define the NetFlow sampling rate for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **sampler name**
3. **description string**
4. **mode {deterministic {m - n} | random {m - n}}**
5. **end**
6. **show sampler [name]**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	sampler <i>name</i> Example: <pre>Switch(config)# sampler SampleTest</pre>	Creates a sampler and enters flow sampler configuration mode.
Step 3	description <i>string</i> Example: <pre>Switch(config-flow-sampler)# description samples</pre>	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	mode {deterministic {<i>m</i> - <i>n</i>} random {<i>m</i> - <i>n</i>}} Example: <pre>Switch(config-flow-sampler)# mode random 1 out-of-1022</pre>	<p>Defines the random sample mode.</p> <p>You can configure either a random or deterministic sampler to an interface. Select <i>m</i> packets out of an <i>n</i> packet window. The window size to select packets from ranges from 32 to 1022.</p> <p>Note the following when configuring a sampler to an interface:</p> <ul style="list-style-type: none"> • When you attach a monitor using deterministic sampler (for example, s1), every attachment with same sampler s1 uses one new free sampler from the device (hardware) out of 4 available samplers. Therefore, beyond 4 attachments, you are not allowed to attach a monitor with any sampler. • In contrast, when you attach a monitor using random sampler (for example-again, s1), only the first attachment uses a new sampler from the device (hardware). The rest of all attachments using the same sampler s1, share the same sampler. • Due to this behavior, when using a deterministic sampler, you can always make sure the correct number of flows are sampled by comparing the sampling rate and what the device sends. If the same random sampler is used with multiple interfaces, flows from an interface can always be sampled, and the flows from other interfaces could be always skipped.

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config-flow-sampler)# end</pre>	Returns to privileged EXEC mode.
Step 6	show sampler [<i>name</i>] Example: <pre>Switch show sample SampleTest</pre>	(Optional) Displays information about NetFlow samplers.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Apply the flow monitor to a source interface or a SVI.

Applying a Flow to an Interface

You can apply a flow monitor and an optional sampler to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. **{ip flow monitor | ipv6 flow monitor}** *name* [**sampler** *name*] **{input}**
4. **end**
5. **show flow interface** [*interface-type number*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>type</i> Example:	Enters interface configuration mode and configures an interface.

	Command or Action	Purpose
	<code>Switch(config)# interface GigabitEthernet1/0/1</code>	
Step 3	<p><code>{ip flow monitor ipv6 flow monitor}name [sampler name] {input}</code></p> <p>Example:</p> <pre>Switch(config-if)# ip flow monitor MonitorTest input</pre>	<p>Associate an IPv4 or an IPv6 flow monitor, and an optional sampler to the interface for input or output packets.</p> <p>You can associate multiple monitors to an interface in both input and output directions.</p>
Step 4	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config-flow-monitor)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p><code>show flow interface [interface-type number]</code></p> <p>Example:</p> <pre>Switch# show flow interface</pre>	(Optional) Displays information about NetFlow on an interface.
Step 6	<p><code>copy running-config startup-config</code></p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Bridged NetFlow on a VLAN

You can apply a flow monitor and an optional sampler to a VLAN.

SUMMARY STEPS

1. `configure terminal`
2. `vlan [configuration] vlan-id`
3. `ip flow monitor monitor name [sampler sampler name] {input }`
4. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>configure terminal</code></p> <p>Example:</p>	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 2	vlan [configuration] <i>vlan-id</i> Example: Switch(config)# <code>vlan configuration 30</code> Switch(config-vlan-config)#	Enters VLAN or VLAN configuration mode.
Step 3	ip flow monitor <i>monitor name</i> [sampler <i>sampler name</i>] {input } Example: Switch(config-vlan-config)# <code>ip flow monitor</code> MonitorTest input	Associates a flow monitor and an optional sampler to the VLAN for input packets.
Step 4	copy running-config startup-config Example: Switch# <code>copy running-config</code> startup-config	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 NetFlow

You can define Layer 2 keys in NetFlow Lite records that you can use to capture flows in Layer 2 interfaces.

SUMMARY STEPS

1. `configure terminal`
2. `flow record name`
3. `match datalink {ethertype | mac {destination {address input} | source {address input}}}`
4. `end`
5. `show flow record [name]`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	flow record <i>name</i> Example: <pre>Switch(config)# flow record L2_record Switch(config-flow-record)#</pre>	Enters flow record configuration mode.
Step 3	match datalink { <i>ethertype</i> <i>mac</i> { <i>destination</i> { <i>address input</i> } <i>source</i> { <i>address input</i> }}} Example: <pre>Switch(config-flow-record)# match datalink mac source address input Switch(config-flow-record)# match datalink mac destination address input</pre>	Specifies the Layer 2 attribute as a key. In this example, the keys are the source and destination MAC addresses from the packet at input. Note When a datalink flow monitor is assigned to an interface or VLAN record, it only creates flows for non-IPv4 or non-IPv6 traffic.
Step 4	end Example: <pre>Switch(config-flow-record)# end</pre>	Returns to privileged EXEC mode.
Step 5	show flow record [<i>name</i>] Example: <pre>Switch# show flow record</pre>	(Optional) Displays information about NetFlow on an interface.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring Flexible NetFlow

The commands in the following table can be used to monitor Flexible NetFlow.

Table 82: Flexible NetFlow Monitoring Commands

Command	Purpose
show flow exporter [<i>broker</i> <i>export-ids</i> <i>name</i> <i>name</i> <i>statistics</i> <i>templates</i>]	Displays information about NetFlow flow exporters and statistics.

Command	Purpose
show flow exporter [name <i>exporter-name</i>]	Displays information about NetFlow flow exporters and statistics.
show flow interface	Displays information about NetFlow interfaces.
	Displays information about NetFlow flow monitors and statistics.
show flow monitor statistics	Displays the statistics for the flow monitor
	Displays the contents of the cache for the flow monitor, in the format specified.
show flow record [name <i>record-name</i>]	Displays information about NetFlow flow records.
show sampler [broker name <i>name</i>]	Displays information about NetFlow samplers.

Configuration Examples for NetFlow Lite

Example: Configuring a Flow



Note When configuring a flow, you need to have the protocol, source port, destination port, first and last timestamps, and packet and bytes counters defined in the flow record. Otherwise, you will get the following error message: "Warning: Cannot set protocol distribution with this Flow Record. Require protocol, source and destination ports, first and last timestamps and packet and bytes counters."

This example shows how to create a flow and apply it to an interface:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)# flow exporter export1
Switch(config-flow-exporter)# destination 10.0.101.254
Switch(config-flow-exporter)# transport udp 2055
Switch(config-flow-exporter)# template data timeout 60
Switch(config-flow-exporter)# exit
Switch(config)# flow record record1
Switch(config-flow-record)# match ipv4 source address
Switch(config-flow-record)# match ipv4 destination address
Switch(config-flow-record)# match ipv4 protocol
Switch(config-flow-record)# match transport source-port
Switch(config-flow-record)# match transport destination-port
Switch(config-flow-record)# collect counter bytes long
Switch(config-flow-record)# collect counter packets long
Switch(config-flow-record)# collect timestamp sys-uptime first
Switch(config-flow-record)# collect timestamp sys-uptime last
Switch(config-flow-record)# exit
Switch(config)# sampler SampleTest
```

```
Switch(config-sampler)# mode random 1 out-of 100
Switch(config-sampler)# exit
Switch(config)# flow monitor monitor1
Switch(config-flow-monitor)# cache timeout active 300
Switch(config-flow-monitor)# cache timeout inactive 120
Switch(config-flow-monitor)# record record1
Switch(config-flow-monitor)# exporter export1
Switch(config-flow-monitor)# exit
Switch(config)# interface GigabitEthernet1/0/1
Switch(config-if)# ip flow monitor monitor1 sampler SampleTest input
Switch(config-if)# end
```

Feature Information for Flexible NetFlow

Release	Modification
Cisco IOS Release 15.2(3)E	This feature was introduced.
Cisco IOS XE Gibraltar 16.12.1	Support was introduced for SGT and DGT fields over FNF for IPv6 traffic.



CHAPTER 44

Configuring Cache Services Using the Web Cache Communication Protocol

- [Finding Feature Information, on page 833](#)
- [Prerequisites for WCCP, on page 833](#)
- [Restrictions for WCCP, on page 834](#)
- [Information About WCCP, on page 835](#)
- [How to Configure WCCP, on page 838](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for WCCP

Before configuring WCCP on your switch, make sure you adhere to the following configuration prerequisites:

- The application engines and switches in the same service group must be in the same subnetwork directly connected to the switch that has WCCP enabled.
- Configure the switch interfaces that are connected to the clients, the application engines, and the server as Layer 3 interfaces (routed ports and switch virtual interfaces [SVIs]). For WCCP packet redirection to work, the servers, application engines, and clients must be on different subnets.
- Use only nonreserved multicast addresses when configuring a single multicast address for each application engine.
- WCCP entries and PBR entries use the same TCAM region. WCCP is supported only on the templates that support PBR: access, routing, and dual IPv4/v6 routing.
- When TCAM entries are not available to add WCCP entries, packets are not redirected and are forwarded by using the standard routing tables.

- The number of available policy-based routing (PBR) labels are reduced as more interfaces are enabled for WCCP ingress redirection. For every interface that supports service groups, one label is consumed. The WCCP labels are taken from the PBR labels. You need to monitor and manage the labels that are available between PBR and WCCP. When labels are not available, the switch cannot add service groups. However, if another interface has the same sequence of service groups, a new label is not needed, and the group can be added to the interface.
- The routing maximum transmission unit (MTU) size configured on the stack member switches should be larger than the client MTU size. The MAC-layer MTU size configured on ports connected to application engines should consider the GRE tunnel header bytes.

Restrictions for WCCP

Unsupported WCCP Features

The following WCCP features are not supported in this software release:

- Packet redirection on an outbound interface that is configured by using the **ip wccp redirect out** interface configuration command.
- The GRE forwarding method for packet redirection.
- GRE redirect and return.
- On the Cisco Catalyst 3650-CX switches, to avoid packet loss you must use the flow control interface configuration command on the 1 gigabyte port connected to the Customer Edge (CE).
- WCCP over GRE
- The hash assignment method for load balancing.
- SNMP support for WCCP.
- Hash assignments in hardware. You can load balance using mask assignments only.
- Redirection for fragmented packets. This is a security feature.
- WCCP with multicast.

General Restrictions

- Maximum number of service groups: eight ingress and eight egress.
- You cannot configure WCCP and VPN routing/forwarding (VRF) on the same switch interface.
- You cannot configure WCCP and PBR on the same switch interface.
- You cannot configure WCCP and a private VLAN (PVLAN) on the same switch interface.
- The **ip wccp redirect exclude in** command allows you to exclude ingress packets from egress WCCP methods. It is not needed on the interface to CE.
- When no cache engine is available, matching packets are dropped. This is closed group support. There is no VRF-aware WCCP support and no IPv6 WCCP.

- When the device is configured with the **ip wccp check services all** command, if the redirect ACL fails to match on packet, it will be checked against the next priority service group.

Information About WCCP

WCCP Overview



Note To use this feature, the device must be running the IP Services feature set.

WCCP is supported only on Cisco Catalyst 3560-CX switches.

WCCP is a Cisco-developed content-routing technology that you can use to integrate wide-area application engines (referred to as application engines) into your network infrastructure. The application engines transparently store frequently accessed content and then fulfill successive requests for the same content, eliminating repetitive transmissions of identical content from servers. Application engines accelerate content delivery and ensure maximum scalability and availability of content. In a service-provider network, you can deploy the WCCP and application engine solution at the points of presence (POPs). In an enterprise network, you can deploy the WCCP and application engine solution at a regional site or small branch office.

The WCCP and Cisco cache engines (or other application engines running WCCP) localize traffic patterns in the network, enabling content requests to be fulfilled locally.

WCCP enables supported Cisco routers and devices to transparently redirect content requests. With transparent redirection, users do not have to configure their browsers to use a web proxy. Instead, they can use the target URL to request content, and their requests are automatically redirected to an application engine. The word transparent means that the end user does not know that a requested file (such as a web page) came from the application engine instead of from the originally specified server.

When an application engine receives a request, it attempts to service it from its own local cache. If the requested information is not present, the application engine sends a separate request to the end server to retrieve the requested information. After receiving the requested information, the application engine forwards it to the requesting client and also caches it to fulfill future requests.

With WCCP, the application-engine cluster (a series of application engines) can service multiple routers or devices.

WCCP Message Exchange

The following sequence of events describes the WCCP message exchange:

1. The application engines send their IP addresses to the WCCP-enabled device by using WCCP, signaling their presence through a Here I am message. The device and application engines communicate to each other through a control channel based on UDP port 2048.
2. The WCCP-enabled device uses the application engine IP information to create a cluster view (a list of application engines in the cluster). This view is sent through an I see you message to each application engine in the cluster, essentially making all the application engines aware of each other. A stable view is established after the membership of the cluster remains the same for a certain amount of time.

3. When a stable view is established, the application engine in the cluster with the lowest IP address is elected as the designated application engine.

WCCP Negotiation

In the exchange of WCCP protocol messages, the designated application engine and the WCCP-enabled device negotiate these items:

- Forwarding method (the method by which the device forwards packets to the application engine). The device rewrites the Layer 2 header by replacing the packet destination MAC address with the target application engine MAC address. It then forwards the packet to the application engine. This forwarding method requires the target application engine to be directly connected to the device at Layer 2.
- Assignment method (the method by which packets are distributed among the application engines in the cluster). The device uses some bits of the destination IP address, the source IP address, the destination Layer 4 port, and the source Layer 4 port to determine which application engine receives the redirected packets.
- Packet-return method (the method by which packets are returned from the application engine to the device for normal forwarding). These are the typical reasons why an application engine rejects packets and starts the packet-return feature:
 - The application engine is overloaded and has no room to service the packets.
 - The application engine receives an error message (such as a protocol or authentication error) from the server and uses the dynamic client bypass feature. The bypass enables clients to bypass the application engines and to connect directly to the server.

The application engine returns a packet to the WCCP-enabled device to forward to the server as if the application engine is not present. The application engine does not intercept the reconnection attempt. In this way, the application engine effectively cancels the redirection of a packet to the application engine and creates a bypass flow. If the return method is Layer 2 rewrite, the packets are forwarded in hardware to the target server. When the server responds with the information, the device uses normal Layer 3 forwarding to return the information to the requesting client.

MD5 Security

WCCP provides an optional security component in each protocol message to enable the device to use MD5 authentication on messages between the device and the application engine. Messages that do not authenticate by MD5 (when authentication of the device is enabled) are discarded by the device. The password string is combined with the MD5 value to create security for the connection between the device and the application engine. You must configure the same password on each application engine.

Packet Redirection and Service Groups

You can configure WCCP to classify traffic for redirection, such as FTP, proxy-web-cache handling, and audio and video applications. This classification, known as a service group, is based on the protocol type (TCP or UDP) and the Layer 4 source destination port numbers. The service groups are identified either by well-known names such as web-cache, which means TCP port 80, or a service number, 0 to 99. Service groups are configured to map to a protocol and Layer 4 port numbers and are established and maintained independently.

WCCP allows dynamic service groups, where the classification criteria are provided dynamically by a participating application engine.

You can configure up to 8 service groups on a device or device stack and up to 32 cache engines per service group. WCCP maintains the priority of the service group in the group definition. WCCP uses the priority to configure the service groups in the device hardware. For example, if service group 1 has a priority of 100 and looks for destination port 80, and service group 2 has a priority of 50 and looks for source port 80, the incoming packet with source and destination port 80 is forwarded by using service group 1 because it has the higher priority.

WCCP supports a cluster of application engines for every service group. Redirected traffic can be sent to any one of the application engines. The device supports the mask assignment method of load balancing the traffic among the application engines in the cluster for a service group.

After WCCP is configured on the device, the device forwards all service group packets received from clients to the application engines. However, the following packets are not redirected:

- Packets originating from the application engine and targeted to the server.
- Packets originating from the application engine and targeted to the client.
- Packets returned or rejected by the application engine. These packets are sent to the server.

You can configure a single multicast address per service group for sending and receiving protocol messages. When there is a single multicast address, the application engine sends a notification to one address, which provides coverage for all routers in the service group, for example, 225.0.0.0. If you add and remove routers dynamically, using a single multicast address provides easier configuration because you do not need to specifically enter the addresses of all devices in the WCCP network.

You can use a router group list to validate the protocol packets received from the application engine. Packets matching the address in the group list are processed, packets not matching the group list address are dropped.

To disable caching for specific clients, servers, or client/server pairs, you can use a WCCP redirect access control list (ACL). Packets that do not match the redirect ACL bypass the cache and are forwarded normally.

Before WCCP packets are redirected, the device examines ACLs associated with all inbound features configured on the interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL.



Note Both permit and deny ACL entries are supported in WCCP redirect lists.

When packets are redirected, the output ACLs associated with the redirected interface are applied to the packets. Any ACLs associated with the original port are not applied unless you specifically configure the required output ACLs on the redirected interfaces.

How to Configure WCCP

Default WCCP Configuration

Feature	Default Setting
WCCP enable state	WCCP services are disabled.
Protocol version	WCCPv2.
Redirecting traffic received on an interface	Disabled.

Enabling the Cache Service

For WCCP packet redirection to operate, you must configure the device interface connected to the client to redirect inbound packets.

This procedure shows how to configure these features on routed ports. To configure these features on SVIs, see the configuration examples that follow the procedure.

Follow these steps to enable the cache service, to set a multicast group address or group list, to configure routed interfaces, to redirect inbound packets received from a client to the application engine, enable an interface to listen for a multicast address, and to set a password. This procedure is required.

Before you begin

Configure the SDM template, and reboot the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip wccp** {**web-cache** | *service-number*} [**group-address** *groupaddress*] [**group-list** *access-list*] [**redirect-list** *access-list*] [**password** *encryption-number password*]
4. **interface** *interface-id*
5. **no switchport**
6. **ip address** *ip-address subnet-mask*
7. **no shutdown**
8. **exit**
9. **interface** *interface-id*
10. **no switchport**
11. **ip address** *ip-address subnet-mask*
12. **no shutdown**
13. **ip wccp** {**web-cache** | *service-number*} **redirect in**
14. **ip wccp** {**web-cache** | *service-number*} **group-listen**
15. **exit**
16. **end**
17. **show running-config**

18. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip wccp {web-cache <i>service-number</i>} [group-address <i>groupaddress</i>] [group-list <i>access-list</i>] [redirect-list <i>access-list</i>] [password <i>encryption-number password</i>]</p> <p>Example:</p> <pre>Switch(config)# ip wccp web-cache</pre>	<p>Enables the cache service, and specifies the service number that corresponds to a dynamic service that is defined by the application engine. By default, this feature is disabled.</p> <p>(Optional) For group-address <i>groupaddress</i>, specifies the multicast group address used by the devices and the application engines to participate in the service group.</p> <p>(Optional) For group-list <i>access-list</i>, if a multicast group address is not used, specify a list of valid IP addresses that correspond to the application engines that are participating in the service group.</p> <p>(Optional) For redirect-list <i>access-list</i>, specify the redirect service for specific hosts or specific packets from hosts.</p> <p>(Optional) For password <i>encryption-number password</i>, specify an encryption number. The range is 0 to 7. Use 0 for not encrypted, and use 7 for proprietary. Specify a password name up to seven characters in length. The device combines the password with the MD5 authentication value to create security for the connection between the device and the application engine. By default, no password is configured, and no authentication is performed.</p> <p>You must configure the same password on each application engine.</p> <p>When authentication is enabled, the device discards messages that are not authenticated.</p>
Step 4	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet1/0/1</pre>	<p>Specifies the interface connected to the application engine or the server, and enters interface configuration mode.</p>

	Command or Action	Purpose
Step 5	no switchport Example: Switch(config-if)# no switchport	Enters Layer 3 mode.
Step 6	ip address ip-address subnet-mask Example: Switch(config-if)# ip address 172.20.10.30 255.255.255.0	Configures the IP address and subnet mask.
Step 7	no shutdown Example: Switch(config-if)# no shutdown	Enables the interface.
Step 8	exit Example: Switch(config-if)# exit	Returns to global configuration mode. Repeat Steps 4 through 8 for each application engine and server.
Step 9	interface interface-id Example: Switch(config)# interface gigabitethernet1/0/2	Specifies the interface connected to the client, and enters interface configuration mode.
Step 10	no switchport Example: Switch(config-if)# no switchport	Enters Layer 3 mode.
Step 11	ip address ip-address subnet-mask Example: Switch(config-if)# ip address 175.20.20.10 255.255.255.0	Configures the IP address and subnet mask.
Step 12	no shutdown Example: Switch(config-if)# no shutdown	Enables the interface.
Step 13	ip wccp {web-cache service-number} redirect in Example: Switch(config-if)# ip wccp web-cache redirect in	Redirects packets received from the client to the application engine. Enable this on the interface connected to the client.

	Command or Action	Purpose
Step 14	ip wccp {web-cache service-number} group-listen Example: Switch(config-if)# ip wccp web-cache group-listen	(Optional) When using a multicast group address, the group-listen keyword enables the interface to listen for the multicast address. Enable this on the interface connected to the application engine.
Step 15	exit Example: Switch(config-if)# exit	Returns to global configuration mode. Repeat Steps 9 through 15 for each client.
Step 16	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 17	show running-config Example: Switch# show running-config	Verifies your entries.
Step 18	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuration Examples

This example shows how to configure routed interfaces and to enable the cache service with a multicast group address and a redirect access list. Gigabit Ethernet port 1 is connected to the application engine, is configured as a routed port with an IP address of 172.20.10.30, and is reenabled. Gigabit Ethernet port 2 is connected through the Internet to the server, is configured as a routed port with an IP address of 175.20.20.10, and is reenabled. Gigabit Ethernet ports 3 to 6 are connected to the clients and are configured as routed ports with IP addresses 175.20.30.20, 175.20.40.30, 175.20.50.40, and 175.20.60.50. The device listens for multicast traffic and redirects packets received from the client interfaces to the application engine.

```
Switch# configure terminal
Switch(config)# ip wccp web-cache group-address 224.1.1.100 redirect list 12
Switch(config)# access-list 12 permit host 10.1.1.1
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.20.10.30 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache group-listen
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
```

```

Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.20.10 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.30.20 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.40.30 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/5
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.50.40 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/6
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.60.50 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit

```

This example shows how to configure SVIs and how to enable the cache service with a multicast group list. VLAN 299 is created and configured with an IP address of 175.20.20.10. Gigabit Ethernet port 1 is connected through the Internet to the server and is configured as an access port in VLAN 299. VLAN 300 is created and configured with an IP address of 172.20.10.30. Gigabit Ethernet port 2 is connected to the application engine and is configured as an access port in VLAN 300. VLAN 301 is created and configured with an IP address of 175.20.30.50. Fast Ethernet ports 3 to 6, which are connected to the clients, are configured as access ports in VLAN 301. The device redirects packets received from the client interfaces to the application engine.



Note Both permit and deny ACL entries are supported in WCCP redirect lists.

```

Switch# configure terminal
Switch(config)# ip wccp web-cache group-list 15
Switch(config)# access-list 15 permit host 171.69.198.102
Switch(config)# access-list 15 permit host 171.69.198.104
Switch(config)# access-list 15 permit host 171.69.198.106
Switch(config)# vlan 299
Switch(config-vlan)# exit
Switch(config)# interface vlan 299
Switch(config-if)# ip address 175.20.20.10 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 299
Switch(config)# vlan 300
Switch(config-vlan)# exit
Switch(config)# interface vlan 300
Switch(config-if)# ip address 171.69.198.100 255.255.255.0
Switch(config-if)# exit

```

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 300
Switch(config-if)# exit
Switch(config)# vlan 301
Switch(config-vlan)# exit
Switch(config)# interface vlan 301
Switch(config-if)# ip address 175.20.30.20 255.255.255.0
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface range gigabitethernet1/0/3 - 6
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 301
Switch(config-if-range)# exit
```

What to do next

To disable the cache service, use the **no ip wccp web-cache** global configuration command. To disable inbound packet redirection, use the **no ip wccp web-cache redirect in** interface configuration command. After completing this procedure, configure the application engines in the network.



PART **VII**

QoS

- [Configuring QoS, on page 847](#)
- [Configuring Auto-QoS, on page 921](#)



CHAPTER 45

Configuring QoS

- [Finding Feature Information, on page 847](#)
- [Prerequisites for QoS, on page 847](#)
- [Restrictions for QoS, on page 849](#)
- [Information About QoS, on page 850](#)
- [How to Configure QoS, on page 870](#)
- [Monitoring Standard QoS, on page 910](#)
- [Configuration Examples for QoS, on page 911](#)
- [Where to Go Next, on page 920](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for QoS

Before configuring standard QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. For example, is the traffic on your network bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

QoS ACL Guidelines

Follow these guidelines when configuring QoS with access control lists (ACLs):

- It is not possible to match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.
- Only one ACL per class map and only one **match** class-map configuration command per class map are supported. The ACL can have multiple ACEs, which match fields against the contents of the packet.
- A trust statement in a policy map requires multiple hardware entries per ACL line. If an input service policy map contains a trust statement in an ACL, the access list might be too large to fit into the available QoS hardware memory, and an error can occur when you apply the policy map to a port. Whenever possible, you should minimize the number of lines in a QoS ACL.

Policing Guidelines

- The port ASIC device, which controls more than one physical port, supports 256 policers (255 user-configurable policers plus 1 policer reserved for system internal use). The maximum number of user-configurable policers supported per port is 63. Policers are allocated on demand by the software and are constrained by the hardware and ASIC boundaries.
You cannot reserve policers per port; there is no guarantee that a port will be assigned to any policer.
- Only one policer is applied to a packet on an ingress port. Only the average rate and committed burst parameters are configurable.
- On a port configured for QoS, all traffic received through the port is classified, policed, and marked according to the policy map attached to the port. On a trunk port configured for QoS, traffic in all VLANs received through the port is classified, policed, and marked according to the policy map attached to the port.
- If you have EtherChannel ports configured on your switch, you must configure QoS classification, policing, mapping, and queueing on the individual physical ports that comprise the EtherChannel. You must decide whether the QoS configuration should match on all ports in the EtherChannel.
- If you need to modify a policy map of an existing QoS policy, first remove the policy map from all interfaces, and then modify or copy the policy map. After you finish the modification, apply the modified policy map to the interfaces. If you do not first remove the policy map from all interfaces, high CPU usage can occur, which, in turn, can cause the console to pause for a very long time.

General QoS Guidelines

These are the general QoS guidelines:

- You configure QoS only on physical ports; there is no support for it at the VLAN level.
- Control traffic (such as spanning-tree bridge protocol data units [BPDUs] and routing update packets) received by the switch are subject to all ingress QoS processing.
- You are likely to lose data when you change queue settings; therefore, try to make changes when traffic is at a minimum.

Restrictions for QoS

The following are the restrictions for QoS:

- To use these features, the switch must be running the LAN Base image: stacking, DSCP, auto-QoS, trusted boundary, policing, marking, mapping tables, and weighted tail drop.
- Ingress queueing is not supported.
- The switch supports 4 default egress queues, with the option to enable an additional 4 egress queues for a total of 8. This option is only available on a standalone switch running the LAN Base image.
- We recommend that you do not enable 8 egress queues by using the **mls qos srr-queue output queues 8** command, when running the following features in your configuration:
 - Auto-QoS
 - Auto SmartPort
 - EnergyWise

Running these features with 8 egress queue enabled in a single configuration is not supported on the switch.

- You can configure QoS only on physical ports. VLAN-based QoS is not supported. You configure the QoS settings, such as classification, queueing, and scheduling, and apply the policy map to a port. When configuring QoS on a physical port, you apply a nonhierarchical policy map to a port.
- If the switch is running the LAN Lite image you can:
 - Configure ACLs, but you cannot attach them to physical interfaces. You can attach them to VLAN interfaces to filter traffic to the CPU.
 - Enable only cos trust at interface level.
 - Enable SRR shaping and sharing at interface level.
 - Enable Priority queueing at interface level.
 - Enable or disable **mls qos rewrite ip dscp**.
- The switch must be running the LAN Base image to use the following QoS features:
 - Policy maps
 - Policing and marking
 - Mapping tables
 - WTD

Information About QoS

QoS Implementation

Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

When you configure the QoS feature, you can select specific network traffic, prioritize it according to its relative importance, and use congestion-management and congestion-avoidance techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation is based on the Differentiated Services (Diff-Serv) architecture, a standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network.

The classification is carried in the IP packet header, using 6 bits from the deprecated IP type of service (ToS) field to carry the classification (*class*) information. Classification can also be carried in the Layer 2 frame.

Figure 75: QoS Classification Layers in Frames and Packets

The special bits in the Layer 2 frame or a Layer 3 packet are shown in the following

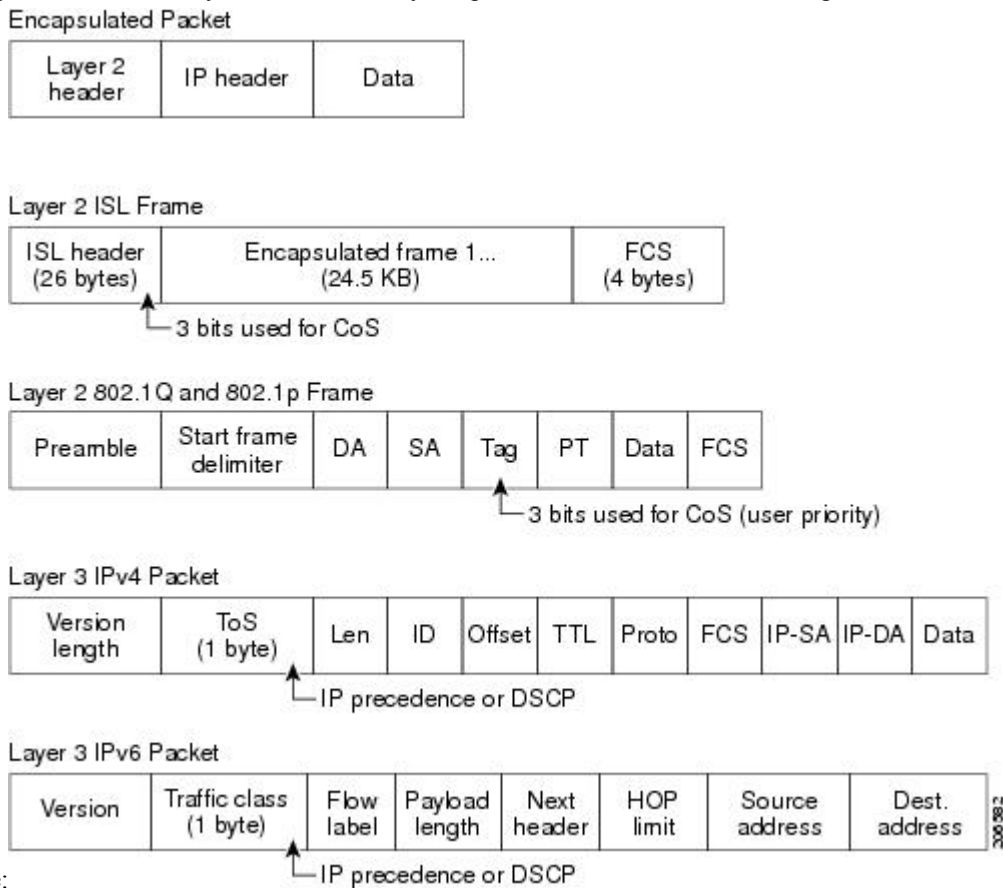


figure:

Layer 2 Frame Prioritization Bits

Layer 2 Inter-Switch Link (ISL) frame headers have a 1-byte User field that carries an IEEE 802.1p class of service (CoS) value in the three least-significant bits. On ports configured as Layer 2 ISL trunks, all traffic is in ISL frames.

Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On ports configured as Layer 2 802.1Q trunks, all traffic is in 802.1Q frames except for traffic in the native VLAN.

Other frame types cannot carry Layer 2 CoS values.

Layer 2 CoS values range from 0 for low priority to 7 for high priority.

Layer 3 Packet Prioritization Bits

Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (DSCP) value. QoS supports the use of either value because DSCP values are backward-compatible with IP precedence values.

IP precedence values range from 0 to 7. DSCP values range from 0 to 63.

End-to-End QoS Solution Using Classification

All switches and routers that access the Internet rely on the class information to provide the same forwarding treatment to packets with the same class information and different treatment to packets with different class information. The class information in the packet can be assigned by end hosts or by switches or routers along the way, based on a configured policy, detailed examination of the packet, or both. Detailed examination of the packet is expected to occur closer to the edge of the network, so that the core switches and routers are not overloaded with this task.

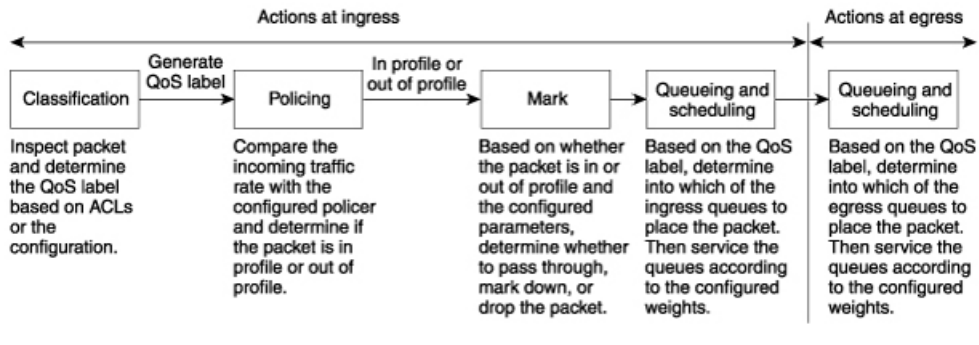
Switches and routers along the path can use the class information to limit the amount of resources allocated per traffic class. The behavior of an individual device when handling traffic in the Diff-Serv architecture is called per-hop behavior. If all devices along a path provide a consistent per-hop behavior, you can construct an end-to-end QoS solution.

Implementing QoS in your network can be a simple task or complex task and depends on the QoS features offered by your internetworking devices, the traffic types and patterns in your network, and the granularity of control that you need over incoming and outgoing traffic.

QoS Basic Model

To implement QoS, the switch must distinguish packets or flows from one another (classify), assign a label to indicate the given quality of service as the packets move through the switch, make the packets comply with the configured resource usage limits (police and mark), and provide different treatment (queue and schedule) in all situations where resource contention exists. The switch also needs to ensure that traffic sent from it meets a specific traffic profile (shape).

Figure 76: QoS Basic Wired Model



Actions at Ingress Port

Actions at the ingress port include classifying traffic, policing, marking, and scheduling:

- Classifying a distinct path for a packet by associating it with a QoS label. The switch maps the CoS or DSCP in the packet to a QoS label to distinguish one kind of traffic from another. The QoS label that is generated identifies all future QoS actions to be performed on this packet.
- Policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result is passed to the marker.
- Marking evaluates the policer and configuration information for the action to be taken when a packet is out of profile and determines what to do with the packet (pass through a packet without modification, marking down the QoS label in the packet, or dropping the packet).



Note Queueing and scheduling are only supported at egress and not at ingress on the switch.

Actions at Egress Port

Actions at the egress port include queueing and scheduling:

- Queueing evaluates the QoS packet label and the corresponding DSCP or CoS value before selecting which of the four egress queues to use. Because congestion can occur when multiple ingress ports simultaneously send data to an egress port, WTD differentiates traffic classes and subjects the packets to different thresholds based on the QoS label. If the threshold is exceeded, the packet is dropped.
- Scheduling services the four egress queues based on their configured SRR shared or shaped weights. One of the queues (queue 1) can be the expedited queue, which is serviced until empty before the other queues are serviced.

Classification Overview

Classification is the process of distinguishing one kind of traffic from another by examining the fields in the packet. Classification is enabled only if QoS is globally enabled on the switch. By default, QoS is globally disabled, so no classification occurs.

During classification, the switch performs a lookup and assigns a QoS label to the packet. The QoS label identifies all QoS actions to be performed on the packet and from which queue the packet is sent.

The QoS label is based on the DSCP or the CoS value in the packet and decides the queuing and scheduling actions to perform on the packet. The label is mapped according to the trust setting and the packet type as shown in the Classification Flowchart.

You specify which fields in the frame or packet that you want to use to classify incoming traffic.

Non-IP Traffic Classification

The following table describes the non-IP traffic classification options for your QoS configuration.

Table 83: Non-IP Traffic Classifications

Non-IP Traffic Classification	Description
Trust the CoS value	Trust the CoS value in the incoming frame (configure the port to trust CoS), and then use the configurable CoS-to-DSCP map to generate a DSCP value for the packet. Layer 2 ISL frame headers carry the CoS value in the 3 least-significant bits of the 1-byte User field. Layer 2 802.1Q frame headers carry the CoS value in the 3 most-significant bits of the Tag Control Information field. CoS values range from 0 for low priority to 7 for high priority.
Trust the DSCP or trust IP precedence value	Trust the DSCP or trust IP precedence value in the incoming frame. These configurations are meaningless for non-IP traffic. If you configure a port with either of these options and non-IP traffic is received, the switch assigns a CoS value and generates an internal DSCP value from the CoS-to-DSCP map. The switch uses the internal DSCP value to generate a CoS value representing the priority of the traffic.
Perform classification based on configured Layer 2 MAC ACL	Perform the classification based on a configured Layer 2 MAC access control list (ACL), which can examine the MAC source address, the MAC destination address, and other fields. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy-map action specifies a DSCP or CoS value to assign to the incoming frame.

After classification, the packet is sent to the policing and marking stages.

IP Traffic Classification

The following table describes the IP traffic classification options for your QoS configuration.

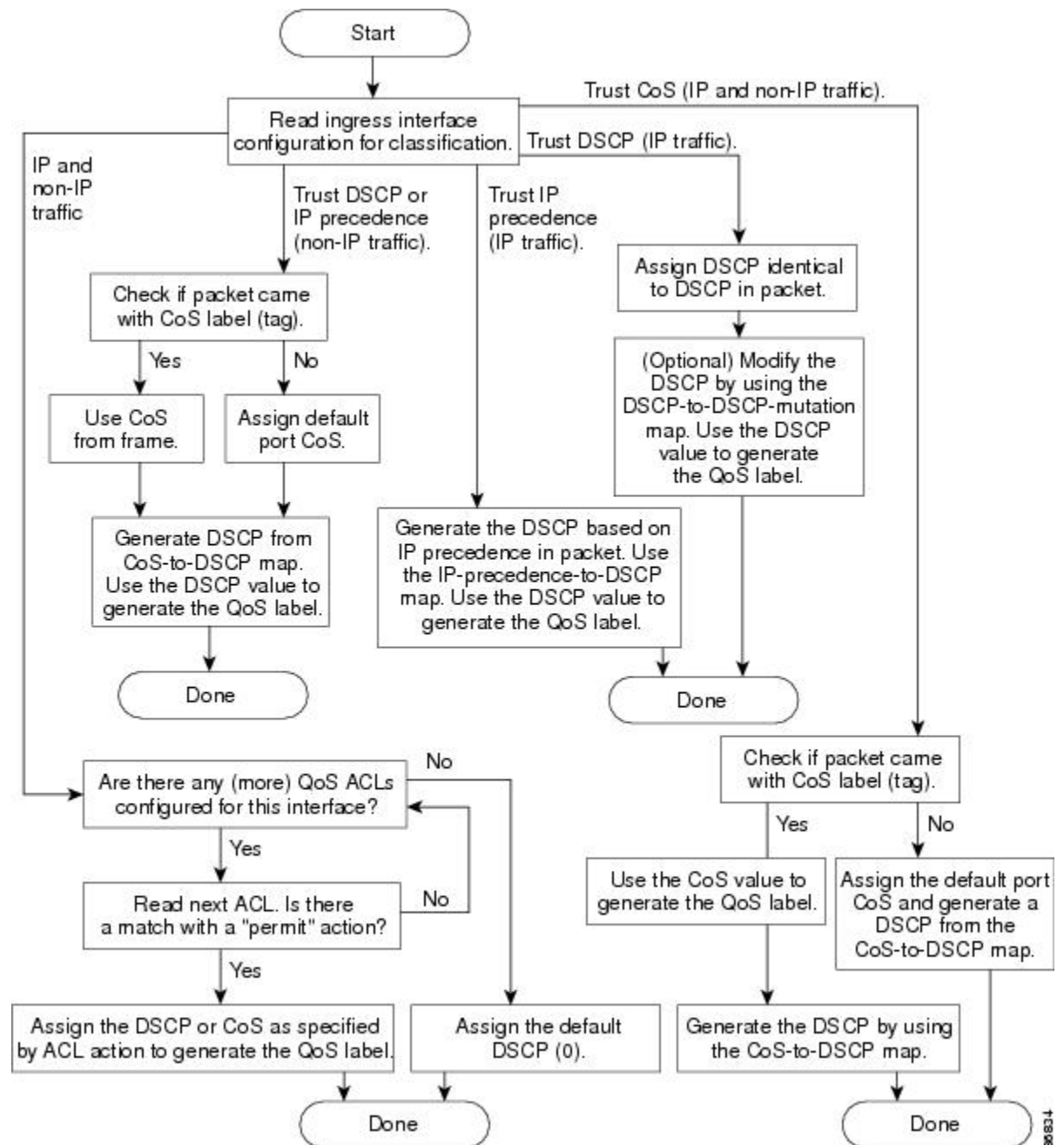
Table 84: IP Traffic Classifications

IP Traffic Classification	Description
Trust the DSCP value	<p>Trust the DSCP value in the incoming packet (configure the port to trust DSCP), and assign the same DSCP value to the packet. The IETF defines the 6 most-significant bits of the 1-byte ToS field as the DSCP. The priority represented by a particular DSCP value is configurable. DSCP values range from 0 to 63.</p> <p>You can also classify IP traffic based on IPv6 DSCP.</p> <p>For ports that are on the boundary between two QoS administrative domains, you can modify the DSCP to another value by using the configurable DSCP-to-DSCP-mutation map.</p>
Trust the IP precedence value	<p>Trust the IP precedence value in the incoming packet (configure the port to trust IP precedence), and generate a DSCP value for the packet by using the configurable IP-precedence-to-DSCP map. The IP Version 4 specification defines the 3 most-significant bits of the 1-byte ToS field as the IP precedence. IP precedence values range from 0 for low priority to 7 for high priority.</p> <p>You can also classify IP traffic based on IPv6 precedence.</p>
Trust the CoS value	<p>Trust the CoS value (if present) in the incoming packet, and generate a DSCP value for the packet by using the CoS-to-DSCP map. If the CoS value is not present, use the default port CoS value.</p>
IP standard or an extended ACL	<p>Perform the classification based on a configured IP standard or an extended ACL, which examines various fields in the IP header. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy-map action specifies a DSCP or CoS value to assign to the incoming frame.</p>
Override configured CoS	<p>Override the configured CoS of incoming packets, and apply the default port CoS value to them. For IPv6 packets, the DSCP value is rewritten by using the CoS-to-DSCP map and by using the default CoS of the port. You can do this for both IPv4 and IPv6 traffic.</p>

After classification, the packet is sent to the policing and marking stages.

Classification Flowchart

Figure 77: Classification Flowchart



Access Control Lists

You can use IP standard, IP extended, or Layer 2 MAC ACLs to define a group of packets with the same characteristics (class). You can also classify IP traffic based on IPv6 ACLs.

In the QoS context, the permit and deny actions in the access control entries (ACEs) have different meanings from security ACLs:

- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.
- If a match with a deny action is encountered, the ACL being processed is skipped, and the next ACL is processed.



Note Deny action is supported in Cisco IOS Release 3.7.4E and later releases.

- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet, and the offers best-effort service to the packet.
- If multiple ACLs are configured on a port, the lookup stops after the packet matches the first ACL with a permit action, and QoS processing begins.



Note When creating an access list, note that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

After a traffic class has been defined with the ACL, you can attach a policy to it. A policy might contain multiple classes with actions specified for each one of them. A policy might include commands to classify the class as a particular aggregate (for example, assign a DSCP) or rate-limit the class. This policy is then attached to a particular port on which it becomes effective.

You implement IP ACLs to classify IP traffic by using the **access-list** global configuration command; you implement Layer 2 MAC ACLs to classify non-IP traffic by using the **mac access-list extended** global configuration command.

Classification Based on Class Maps and Policy Maps

To use policy maps, the switch must be running the LAN Base image.

A class map is a mechanism that you use to name a specific traffic flow (or class) and to isolate it from all other traffic. The class map defines the criteria used to match against a specific traffic flow to further classify it. The criteria can include matching the access group defined by the ACL or matching a specific list of DSCP or IP precedence values. If you have more than one type of traffic that you want to classify, you can create another class map and use a different name. After a packet is matched against the class-map criteria, you further classify it through the use of a policy map.

A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; or specifying the traffic bandwidth limitations and the action to take when the traffic is out of profile. Before a policy map can be effective, you must attach it to a port.

You create a class map by using the **class-map** global configuration command or the **class** policy-map configuration command. You should use the **class-map** command when the map is shared among many ports. When you enter the **class-map** command, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match** class-map configuration command.

You can configure a default class by using the **class class-default** policy-map configuration command. Unclassified traffic (traffic specified in the other traffic classes configured on the policy-map) is treated as default traffic.

You create and name a policy map by using the **policy-map** global configuration command. When you enter this command, the switch enters the policy-map configuration mode. In this mode, you specify the actions to take on a specific traffic class by using the **class**, **trust**, or **set** policy-map configuration and policy-map class configuration commands.

The policy map can contain the **police** and **police aggregate** policy-map class configuration commands, which define the policer, the bandwidth limitations of the traffic, and the action to take if the limits are exceeded.

To enable the policy map, you attach it to a port by using the **service-policy** interface configuration command.

Policing and Marking Overview

After a packet is classified and has a DSCP-based or CoS-based QoS label assigned to it, the policing and marking process can begin.

Policing involves creating a policer that specifies the bandwidth limits for the traffic. Packets that exceed the limits are *out of profile* or *nonconforming*. Each policer decides on a packet-by-packet basis whether the packet is in or out of profile and specifies the actions on the packet. These actions, carried out by the marker, include passing through the packet without modification, dropping the packet, or modifying (marking down) the assigned DSCP of the packet and allowing the packet to pass through. The configurable policed-DSCP map provides the packet with a new DSCP-based QoS label. Marked-down packets use the same queues as the original QoS label to prevent packets in a flow from getting out of order.



Note All traffic, regardless of whether it is bridged or routed, is subjected to a policer, if one is configured. As a result, bridged packets might be dropped or might have their DSCP or CoS fields modified when they are policed and marked.

You can configure policing on a physical port. After you configure the policy map and policing actions, attach the policy to a port by using the **service-policy** interface configuration command.

Physical Port Policing

In policy maps on physical ports, you can create the following types of policers:

- **Individual**—QoS applies the bandwidth limits specified in the policer separately to each matched traffic class. You configure this type of policer within a policy map by using the **police** policy-map class configuration command.
- **Aggregate**—QoS applies the bandwidth limits specified in an aggregate policer cumulatively to all matched traffic flows. You configure this type of policer by specifying the aggregate policer name within a policy map by using the **police aggregate** policy-map class configuration command. You specify the bandwidth limits of the policer by using the **mls qos aggregate-policer** global configuration command. In this way, the aggregate policer is shared by multiple classes of traffic within a policy map.

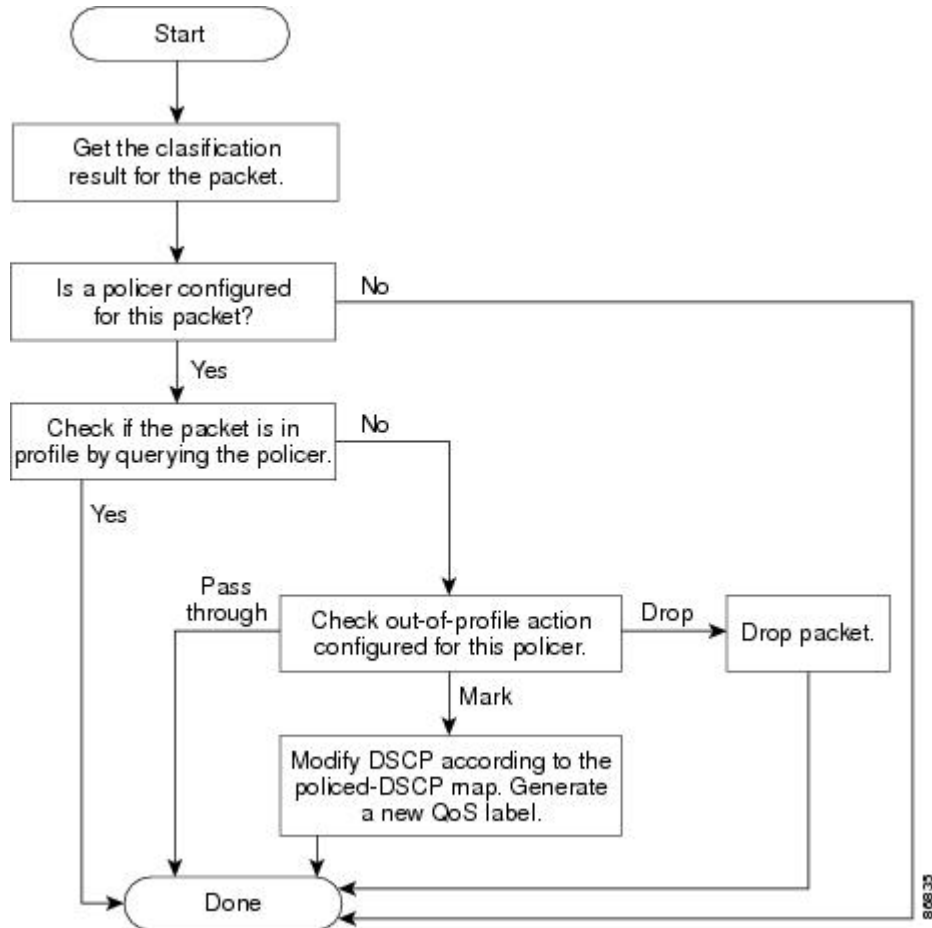
Policing uses a token-bucket algorithm. As each frame is received by the switch, a token is added to the bucket. The bucket has a hole in it and leaks at a rate that you specify as the average traffic rate in bits per second. Each time a token is added to the bucket, the switch verifies that there is enough room in the bucket. If there is not enough room, the packet is marked as nonconforming, and the specified policer action is taken (dropped or marked down).

How quickly the bucket fills is a function of the bucket depth (burst-byte), the rate at which the tokens are removed (rate-bps), and the duration of the burst above the average rate. The size of the bucket imposes an upper limit on the burst length and limits the number of frames that can be transmitted back-to-back. If the

burst is short, the bucket does not overflow, and no action is taken against the traffic flow. However, if a burst is long and at a higher rate, the bucket overflows, and the policing actions are taken against the frames in that burst.

You configure the bucket depth (the maximum burst that is tolerated before the bucket overflows) by using the `burst-byte` option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command. You configure how fast (the average rate) that the tokens are removed from the bucket by using the `rate-bps` option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command.

Figure 78: Policing and Marking Flowchart on Physical Ports



Mapping Tables Overview

During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with a QoS label based on the DSCP or CoS value from the classification stage.

The following table describes QoS processing and mapping tables.

Table 85: QoS Processing and Mapping Tables

QoS Processing Stage	Mapping Table Usage
Classification	<p>During the classification stage, QoS uses configurable mapping tables to derive a corresponding DSCP or CoS value from a received CoS, DSCP, or IP precedence value. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map.</p> <p>You configure these maps by using the mls qos map cos-dscp and the mls qos map ip-prec-dscp global configuration commands.</p> <p>On an ingress port configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, you can apply the configurable DSCP-to-DSCP-mutation map to the port that is on the boundary between the two QoS domains.</p> <p>You configure this map by using the mls qos map dscp-mutation global configuration command.</p>
Policing	<p>During policing stage, QoS can assign another DSCP value to an IP or a non-IP packet (if the packet is out of profile and the policer specifies a marked-down value). This configurable map is called the policed-DSCP map.</p> <p>You configure this map by using the mls qos map policed-dscp global configuration command.</p>
Pre-scheduling	<p>Before the traffic reaches the scheduling stage, QoS stores the packet in an egress queue according to the QoS label. The QoS label is based on the DSCP or the CoS value in the packet and selects the queue through the DSCP output queue threshold maps or through the CoS output queue threshold maps. In addition to an egress queue, the QoS label also identifies the WTD threshold value.</p> <p>You configure these maps by using the mls qos srr-queue { output } dscp-map and the mls qos srr-queue { output } cos-map global configuration commands.</p>

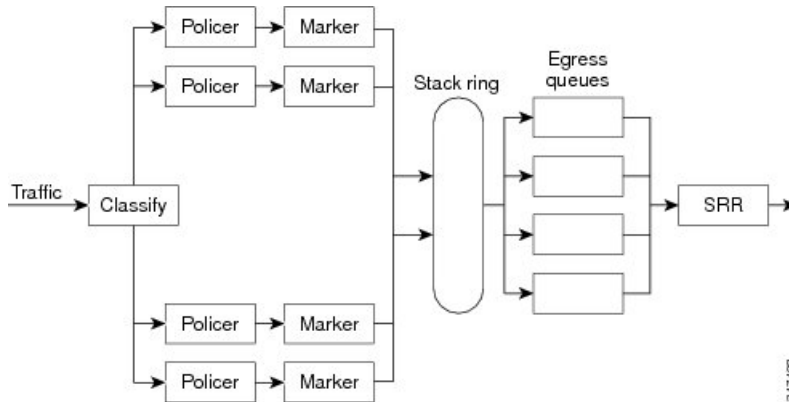
The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP maps have default values that might or might not be appropriate for your network.

The default DSCP-to-DSCP-mutation map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value. The DSCP-to-DSCP-mutation map is the only map you apply to a specific port. All other maps apply to the entire switch.

Queueing and Scheduling Overview

The switch has queues at specific points to help prevent congestion.

Figure 79: Egress Queue Location on Switch



Note The switch supports 4 egress queues by default and there is an option to enable a total of 8 egress queues. The 8 egress queue configuration is only supported on a standalone switch.

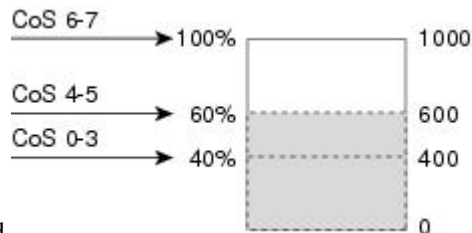
Weighted Tail Drop

As a frame is enqueued to a particular queue, WTD uses the frame's assigned QoS label to subject it to different thresholds. If the threshold is exceeded for that QoS label (the space available in the destination queue is less than the size of the frame), the switch drops the frame.

Each queue has three threshold values. The QoS label determines which of the three threshold values is subjected to the frame. Of the three thresholds, two are configurable (explicit) and one is not (implicit).

Figure 80: WTD and Queue Operation

The following figure shows an example of WTD operating on a queue whose size is 1000 frames. Three drop percentages are configured: 40 percent (400 frames), 60 percent (600 frames), and 100 percent (1000 frames). These percentages indicate that up to 400 frames can be queued at the 40-percent threshold, up to 600 frames at the 60-percent threshold, and up to 1000 frames at the 100-percent



threshold.

In the example, CoS values 6 and 7 have a greater importance than the other CoS values, and they are assigned to the 100-percent drop threshold (queue-full state). CoS values 4 and 5 are assigned to the 60-percent threshold, and CoS values 0 to 3 are assigned to the 40-percent threshold.

Suppose the queue is already filled with 600 frames, and a new frame arrives. It contains CoS values 4 and 5 and is subjected to the 60-percent threshold. If this frame is added to the queue, the threshold will be exceeded, so the switch drops it.

SRR Shaping and Sharing

You can configure SRR on egress queues for sharing or for shaping.

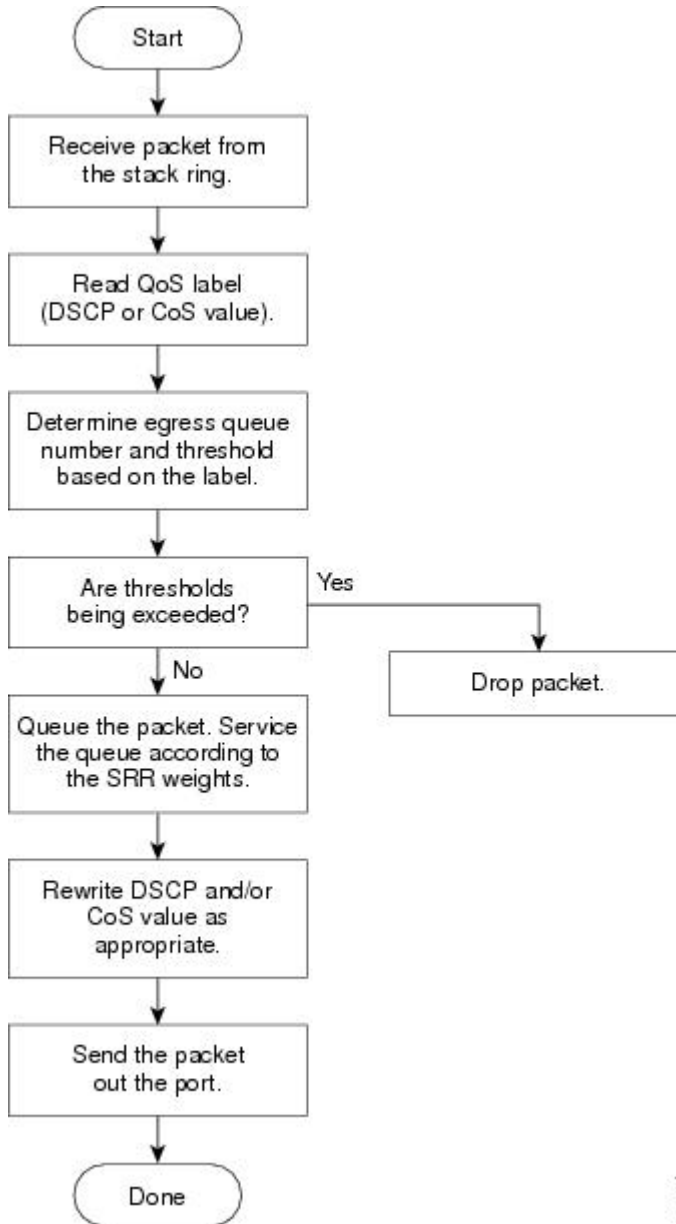
In shaped mode, the egress queues are guaranteed a percentage of the bandwidth, and they are rate-limited to that amount. Shaped traffic does not use more than the allocated bandwidth even if the link is idle. Shaping provides a more even flow of traffic over time and reduces the peaks and valleys of bursty traffic. With shaping, the absolute value of each weight is used to compute the bandwidth available for the queues.

In shared mode, the queues share the bandwidth among them according to the configured weights. The bandwidth is guaranteed at this level but not limited to it. For example, if a queue is empty and no longer requires a share of the link, the remaining queues can expand into the unused bandwidth and share it among them. With sharing, the ratio of the weights controls the frequency of dequeuing; the absolute values are meaningless. Shaping and sharing is configured per interface. Each interface can be uniquely configured.

Queueing and Scheduling on Egress Queues

The following figure shows queueing and scheduling flowcharts for egress ports on the switch.

Figure 81: Queueing and Scheduling Flowchart for Egress Ports on the Switch



Note If the expedite queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Expedite Queue

Each port supports four egress queues, one of which (queue 1) can be the egress expedite queue. These queues are assigned to a queue-set. All traffic exiting the switch flows through one of these four queues and is subjected to a threshold based on the QoS label assigned to the packet.



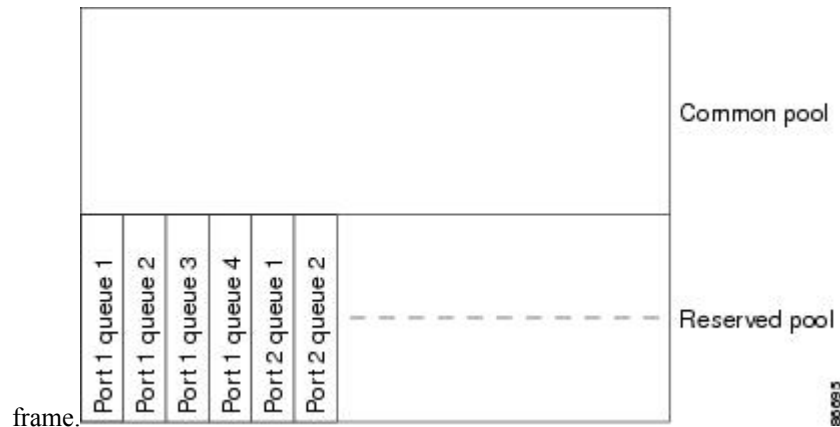
Note If the expedite queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Queue Buffer Allocation

The following figure shows the egress queue buffer.

Figure 82: Egress Queue Buffer Allocation

The buffer space is divided between the common pool and the reserved pool. The switch uses a buffer allocation scheme to reserve a minimum amount of buffers for each egress queue, to prevent any queue or port from consuming all the buffers and depriving other queues, and to control whether to grant buffer space to a requesting queue. The switch detects whether the target queue has not consumed more buffers than its reserved amount (under-limit), whether it has consumed all of its maximum buffers (over limit), and whether the common pool is empty (no free buffers) or not empty (free buffers). If the queue is not over-limit, the switch can allocate buffer space from the reserved pool or from the common pool (if it is not empty). If there are no free buffers in the common pool or if the queue is over-limit, the switch drops the



Buffer and Memory Allocation

You guarantee the availability of buffers, set drop thresholds, and configure the maximum memory allocation for a queue-set by using the **mls qos queue-set output *qset-id* threshold *queue-id* drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold** global configuration command. Each threshold value is a percentage of the queue's allocated memory, which you specify by using the **mls qos queue-set output *qset-id* buffers allocation1 ... allocation4** global configuration command. The sum of all the allocated buffers represents the reserved pool, and the remaining buffers are part of the common pool.

Through buffer allocation, you can ensure that high-priority traffic is buffered. For example, if the buffer space is 400, you can allocate 70 percent of it to queue 1 and 10 percent to queues 2 through 4. Queue 1 then has 280 buffers allocated to it, and queues 2 through 4 each have 40 buffers allocated to them.

You can guarantee that the allocated buffers are reserved for a specific queue in a queue-set. For example, if there are 100 buffers for a queue, you can reserve 50 percent (50 buffers). The switch returns the remaining 50 buffers to the common pool. You also can enable a queue in the full condition to obtain more buffers than are reserved for it by setting a maximum threshold. The switch can allocate the needed buffers from the common pool if the common pool is not empty.



Note The switch supports 4 egress queues by default, although there is an option to enable a total of 8 egress queues. Use the **mls qos srr-queue output queues 8** global configuration command to enable all 8 egress queues. Once 8 egress queues are enabled, you are able to configure thresholds and buffers for all 8 queues. The 8 egress queue configuration is only supported on a standalone switch.

Queues and WTD Thresholds

You can assign each packet that flows through the switch to a queue and to a threshold.

Specifically, you map DSCP or CoS values to an egress queue and map DSCP or CoS values to a threshold ID. You use the **mls qos srr-queue output dscp-map queue** *queue-id* {*dscp1...dscp8* | **threshold** *threshold-id* *dscp1...dscp8*} or the **mls qos srr-queue output cos-map queue** *queue-id* {*cos1...cos8* | **threshold** *threshold-id* *cos1...cos8*} global configuration command. You can display the DSCP output queue threshold map and the CoS output queue threshold map by using the **show mls qos maps** privileged EXEC command.

The queues use WTD to support distinct drop percentages for different traffic classes. Each queue has three drop thresholds: two configurable (*explicit*) WTD thresholds and one nonconfigurable (*implicit*) threshold preset to the queue-full state. You assign the two WTD threshold percentages for threshold ID 1 and ID 2. The drop threshold for threshold ID 3 is preset to the queue-full state, and you cannot modify it. You map a port to queue-set by using the **queue-set qset-id** interface configuration command. Modify the queue-set configuration to change the WTD threshold percentages.



Note The switch supports 4 egress queues by default, although there is an option to enable a total of 8 egress queues. Use the **mls qos srr-queue output queues 8** global configuration command to enable all 8 egress queues. Once 8 egress queues are enabled, you are able to configure thresholds and buffers for all 8 queues. The 8 egress queue configuration is only supported on a standalone switch.

Shaped or Shared Mode

SRR services each queue-set in shared or shaped mode. You map a port to a queue-set by using the **queue-set** *qset-id* interface configuration command.

You assign shared or shaped weights to the port by using the **srr-queue bandwidth share** *weight1 weight2 weight3 weight4* or the **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* interface configuration command.

The buffer allocation together with the SRR weight ratios control how much data can be buffered and sent before packets are dropped. The weight ratio is the ratio of the frequency in which the SRR scheduler sends packets from each queue.

All four queues participate in the SRR unless the expedite queue is enabled, in which case the first bandwidth weight is ignored and is not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular DSCPs or CoSs into certain queues, by allocating a large queue size or by servicing the queue more frequently, and by adjusting queue thresholds so that packets with lower priorities are dropped.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.



Note The switch supports 4 egress queues by default, although there is an option to enable a total of 8 egress queues. Use the **mls qos srr-queue output queues 8** global configuration command to enable all 8 egress queues. Once 8 egress queues are enabled, you are able to configure thresholds, buffers, bandwidth share weights, and bandwidth shape weights for all 8 queues. The 8 egress queue configuration is only supported on a standalone switch.

Packet Modification

A packet is classified, policed, and queued to provide QoS. The following packet modifications can occur during the process to provide QoS:

- For IP and non-IP packets, classification involves assigning a QoS label to a packet based on the DSCP or CoS of the received packet. However, the packet is not modified at this stage; only an indication of the assigned DSCP or CoS value is carried along.
- During policing, IP and non-IP packets can have another DSCP assigned to them (if they are out of profile and the policer specifies a markdown DSCP). Once again, the DSCP in the packet is not modified, but an indication of the marked-down value is carried along. For IP packets, the packet modification occurs at a later stage; for non-IP packets the DSCP is converted to CoS and used for queueing and scheduling decisions.
- Depending on the QoS label assigned to a frame and the mutation chosen, the DSCP and CoS values of the frame are rewritten. If you do not configure a table map and if you configure the port to trust the DSCP of the incoming frame, the DSCP value in the frame is not changed, but the CoS is rewritten according to the DSCP-to-CoS map. If you configure the port to trust the CoS of the incoming frame and it is an IP packet, the CoS value in the frame is not changed, but the DSCP might be changed according to the CoS-to-DSCP map.

The input mutation causes the DSCP to be rewritten depending on the new value of DSCP chosen. The set action in a policy map also causes the DSCP to be rewritten.

Standard QoS Default Configuration

Standard QoS is disabled by default.

When QoS is disabled, there is no concept of trusted or untrusted ports because the packets are not modified. The CoS, DSCP, and IP precedence values in the packet are not changed.

Traffic is switched in pass-through mode. The packets are switched without any rewrites and classified as best effort without any policing.

When QoS is enabled using the **mls qos** global configuration command and all other QoS settings are at their defaults, traffic is classified as best effort (the DSCP and CoS value is set to 0) without any policing. No policy maps are configured. The default port trust state on all ports is untrusted.



Note Starting Cisco IOS Release 15.2(1)E, IPv6 QoS is supported on switches running the LAN base license with lanbase-routing template.

Default Egress Queue Configuration

The following tables describe the default egress queue configurations.



Note The switch supports 4 egress queues by default, although there is an option to enable a total of 8 egress queues. Use the **mls qos srr-queue output queues 8** global configuration command to enable all 8 egress queues. Once 8 egress queues are enabled, you are able to configure thresholds and buffers for all 8 queues. The 8 egress queue configuration is only supported on a standalone switch.

The following table shows the default egress queue configuration for each queue-set when QoS is enabled. All ports are mapped to queue-set 1. The port bandwidth limit is set to 100 percent and rate unlimited. Note that for the SRR shaped weights (absolute) feature, a shaped weight of zero indicates that the queue is operating in shared mode. Note that for the SRR shared weights feature, one quarter of the bandwidth is allocated to each queue.

Table 86: Default Egress Queue Configuration

Feature	Queue 1	Queue 2	Queue 3	Queue 4
Buffer allocation	25 percent	25 percent	25 percent	25 percent
WTD drop threshold 1	100 percent	200 percent	100 percent	100 percent
WTD drop threshold 2	100 percent	200 percent	100 percent	100 percent
Reserved threshold	50 percent	50 percent	50 percent	50 percent
Maximum threshold	400 percent	400 percent	400 percent	400 percent
SRR shaped weights (absolute)	25	0	0	0
SRR shared weights	25	25	25	25

The following table shows the default CoS output queue threshold map when QoS is enabled.

Table 87: Default CoS Output Queue Threshold Map

CoS Value	Queue ID–Threshold ID
0, 1	2–1
2, 3	3–1
4	4–1

CoS Value	Queue ID–Threshold ID
5	1–1
6, 7	4–1

The following table shows the default DSCP output queue threshold map when QoS is enabled.

Table 88: Default DSCP Output Queue Threshold Map

DSCP Value	Queue ID–Threshold ID
0–15	2–1
16–31	3–1
32–39	4–1
40–47	1–1
48–63	4–1

The following table displays the default egress queue configuration when the 8 egress queue configuration is enabled using the **mls qos srr-queue output queues 8** command.

Table 89: Default 8 Egress Queue Configuration

Feature	Queue 1	Queue 2	Queue 3	Queue 4	Queue 5	Queue 6	Queue 7	Queue 8
Buffer allocation	10	30	10	10	10	10	10	10
WTD drop threshold 1	100	1600	100	100	100	100	100	100
WTD drop threshold 2	100	2000	100	100	100	100	100	100
Reserved threshold	100	100	100	100	100	100	100	100
Maximum threshold	400	2400	400	400	400	400	400	400
SRR shaped weights	25	0	0	0	0	0	0	0

Feature	Queue 1	Queue 2	Queue 3	Queue 4	Queue 5	Queue 6	Queue 7	Queue 8
SRR shared weights	25	25	25	25	25	25	25	25

The following table displays the default CoS output queue threshold map when QoS is enabled and the 8 egress queue configuration is enabled using the **mls qos srr-queue output queues 8** command.

Table 90: Default CoS Output 8 Queue Threshold Map

CoS	Egress Queue	Threshold ID	4 Egress Queue Mapping
0	2	1	2
1	3	1	2
2	4	1	3
3	5	1	3
4	6	1	4
5	1	1	1
6	7	1	4
7	8	1	4

The following table displays the default DSCP output queue threshold map when QoS is enabled and the 8 egress queue configuration is enabled using the **mls qos srr-queue output queues 8** command.

Table 91: Default DSCP Output 8 Queue Threshold Map

DSCP	Egress Queue	Threshold ID	4 Egress Queue Mapping
0-7	2	1	2
8-15	3	1	2
16-23	4	1	3
24-31	5	1	3
32-39	6	1	4
40-47	1	1	1
48-55	7	1	4
56-63	8	1	4

Default Mapping Table Configuration

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value (no markdown).

DSCP Maps

Default CoS-to-DSCP Map

When DSCP transparency mode is disabled, the DSCP values are derived from CoS as per the following table. If these values are not appropriate for your network, you need to modify them.

Note The DSCP transparency mode is disabled by default. If it is enabled (**no mls qos rewrite ip dscp** interface configuration command), DSCP rewrite will not happen.

Table 92: Default CoS-to-DSCP Map

CoS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Default IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic. The following table shows the default IP-precedence-to-DSCP map. If these values are not appropriate for your network, you need to modify them.

Table 93: Default IP-Precedence-to-DSCP Map

IP Precedence Value	DSCP Value
0	0
1	8
2	16

IP Precedence Value	DSCP Value
3	24
4	32
5	40
6	48
7	56

Default DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues. The following table shows the default DSCP-to-CoS map. If these values are not appropriate for your network, you need to modify them.

Table 94: Default DSCP-to-CoS Map

DSCP Value	CoS Value
0–7	0
8–15	1
16–23	2
24–31	3
32–39	4
40–47	5
48–55	6
56–63	7

How to Configure QoS

Enabling QoS Globally

By default, QoS is disabled on the switch.

The following procedure to enable QoS globally is required.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **end**

4. `show mls qos`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	mls qos Example: <pre>Switch(config)# mls qos</pre>	Enables QoS globally. QoS operates with the default settings described in the related topic sections below. Note To disable QoS, use the no mls qos global configuration command.
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	show mls qos Example: <pre>Switch# show mls qos</pre>	Verifies the QoS configuration.
Step 5	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling VLAN-Based QoS on Physical Ports

By default, VLAN-based QoS is disabled on all physical switch ports. You can enable VLAN-based QoS on a switch port.

SUMMARY STEPS

1. `configure terminal`
2. `interface interface-id`
3. `mls qos vlan-based`
4. `end`

5. `show mls qos interface interface-id`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the physical port, and enter interface configuration mode.
Step 3	mls qos vlan-based Example: Switch(config-if)# <code>mls qos vlan-based</code>	Enables VLAN-based QoS on the port. Note Use the <code>no mls qos vlan-based</code> interface configuration command to disable VLAN-based QoS on the physical port.
Step 4	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show mls qos interface interface-id Example: Switch# <code>show mls qos interface gigabitethernet 1/0/1</code>	Verifies if VLAN-based QoS is enabled on the physical port.
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a QoS Policy

Configuring a QoS policy typically requires the following tasks:

- Classifying traffic into classes

- Configuring policies applied to those traffic classes
- Attaching policies to ports

These sections describe how to classify, police, and mark traffic. Depending on your network configuration, you must perform one or more of the modules in this section.

Classifying Traffic by Using ACLs

You can classify IP traffic by using IPv4 standard ACLs, IPv4 extended ACLs, or IPv6 ACLs.

You can classify non-IP traffic by using Layer 2 MAC ACLs.

Creating an IP Standard ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>] Example: <pre>Switch(config)# access-list 1 permit 192.2.255.0 10.1.1.255</pre>	Creates an IP standard ACL, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number. The range is 1 to 99 and 1300 to 1999. • Use the permit keyword to permit a certain type of traffic if the conditions are matched. Use the deny keyword to deny a certain type of traffic if conditions are matched. • For <i>source</i>, enter the network or host from which the packet is being sent. You can use the any keyword as an abbreviation for 0.0.0.0 255.255.255.255. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the

	Command or Action	Purpose
		<p>source. Place ones in the bit positions that you want to ignore.</p> <p>When you create an access list, remember that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.</p> <p>Note To delete an access list, use the no access-list access-list-number global configuration command.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Switch(config) # end</pre>	Returns to privileged EXEC mode.
Step 4	<p>show access-lists</p> <p>Example:</p> <pre>Switch# show access-lists</pre>	Verifies your entries.
Step 5	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating an IP Extended ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list access-list-number {deny | permit} protocol source source-wildcard destination destination-wildcard**
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>access-list <i>access-list-number</i> {deny permit} <i>protocol</i> <i>source source-wildcard destination destination-wildcard</i></p> <p>Example:</p> <pre>Switch(config)# access-list 100 permit ip any any dscp 32</pre>	<p>Creates an IP extended ACL, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number. The range is 100 to 199 and 2000 to 2699. • Use the permit keyword to permit a certain type of traffic if the conditions are matched. Use the deny keyword to deny a certain type of traffic if conditions are matched. • For <i>protocol</i>, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocol keywords. • For <i>source</i>, enter the network or host from which the packet is being sent. You specify this by using dotted decimal notation, by using the any keyword as an abbreviation for <i>source</i> 0.0.0.0 <i>source-wildcard</i> 255.255.255.255, or by using the host keyword for <i>source</i> 0.0.0.0. • For <i>source-wildcard</i>, enter the wildcard bits by placing ones in the bit positions that you want to ignore. You specify the wildcard by using dotted decimal notation, by using the any keyword as an abbreviation for <i>source</i> 0.0.0.0 <i>source-wildcard</i> 255.255.255.255, or by using the host keyword for <i>source</i> 0.0.0.0. • For <i>destination</i>, enter the network or host to which the packet is being sent. You have the same options for specifying the <i>destination</i> and <i>destination-wildcard</i> as those described by <i>source</i> and <i>source-wildcard</i>. <p>When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.</p> <p>Note To delete an access list, use the no access-list <i>access-list-number</i> global configuration command.</p>

	Command or Action	Purpose
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 4	show access-lists Example: Switch# show access-lists	Verifies your entries.
Step 5	copy running-config startup-config Example: Switch# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Creating an IPv6 ACL for IPv6 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 access-list** *access-list-name*
3. **{deny | permit} protocol {source-ipv6-prefix/prefix-length | any | host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/ prefix-length | any | host destination-ipv6-address} [operator [port-number]] [dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]**
4. **end**
5. **show ipv6 access-list**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ipv6 access-list <i>access-list-name</i> Example: Switch(config)# ipv6	Creates an IPv6 ACL and enters IPv6 access-list configuration mode. Accesses list names cannot contain a space or quotation mark or begin with a numeric.

	Command or Action	Purpose
	<code>access-list ipv6_Name_ACL</code>	Note To delete an access list, use the no ipv6 access-list access-list-number global configuration command.
Step 3	<p><code>{deny permit} protocol {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/ prefix-length any host destination-ipv6-address} [operator [port-number]] [dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]</code></p> <p>Example:</p> <pre>Switch(config-ipv6-acl)# permit ip host 10::1 host 11::2 host</pre>	<p>Enters deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions:</p> <p>For <i>protocol</i>, enter the name or number of an Internet protocol: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number.</p> <ul style="list-style-type: none"> • The <i>source-ipv6-prefix/prefix-length</i> or <i>destination-ipv6-prefix/ prefix-length</i> is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373). • Enter any as an abbreviation for the IPv6 prefix <code>::/0</code>. • For host <i>source-ipv6-address</i> or <i>destination-ipv6-address</i>, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For <i>operator</i>, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range. If the operator follows the <i>source-ipv6-prefix/prefix-length</i> argument, it must match the source port. If the operator follows the <i>destination-ipv6- prefix/prefix-length</i> argument, it must match the destination port. • (Optional) The <i>port-number</i> is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is IPv6.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) Enter log to cause a logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295. • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 4	end Example: <pre>Switch(config-ipv6-acl) # end</pre>	Returns to privileged EXEC mode.
Step 5	show ipv6 access-list Example: <pre>Switch# show ipv6 access-list</pre>	Verifies the access list configuration.
Step 6	copy running-config startup-config Example: <pre>Switch# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating a Layer 2 MAC ACL for Non-IP Traffic

Before you begin

Before you perform this task, determine that Layer 2 MAC access lists are required for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **mac access-list extended name**
3. **{permit | deny} {host src-MAC-addr mask | any | host dst-MAC-addr | dst-MAC-addr mask} [type mask]**
4. **end**
5. **show access-lists [access-list-number | access-list-name]**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	mac access-list extended name Example: <pre>Switch(config)# mac access-list extended maclist1</pre>	<p>Creates a Layer 2 MAC ACL by specifying the name of the list.</p> <p>After entering this command, the mode changes to extended MAC ACL configuration.</p> <p>Note To delete an access list, use the no mac access-list extended access-list-name global configuration command.</p>
Step 3	<pre>{permit deny} {host src-MAC-addr mask any host dst-MAC-addr dst-MAC-addr mask} [type mask]</pre> Example: <pre>Switch(config-ext-macl) # permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0</pre> <pre>Switch(config-ext-macl) # permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp</pre>	<p>Specifies the type of traffic to permit or deny if the conditions are matched, entering the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>src-MAC-addr</i>, enter the MAC address of the host from which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the any keyword as an abbreviation for <i>source</i> 0.0.0, <i>source-wildcard</i> ffff.ffff.ffff, or by using the host keyword for <i>source</i> 0.0.0. For <i>mask</i>, enter the wildcard bits by placing ones in the bit positions that you want to ignore. For <i>dst-MAC-addr</i>, enter the MAC address of the host to which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the any keyword as an abbreviation for <i>source</i> 0.0.0, <i>source-wildcard</i> ffff.ffff.ffff, or by using the host keyword for <i>source</i> 0.0.0. (Optional) For <i>type mask</i>, specify the Ethertype number of a packet with Ethernet II or SNAP encapsulation to identify the protocol of the packet. For <i>type</i>, the range is from 0 to 65535, typically specified in hexadecimal. For <i>mask</i>, enter the <i>don't care</i> bits applied to the Ethertype before testing for a match. <p>When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.</p>

	Command or Action	Purpose
Step 4	end Example: <pre>Switch(config-ext-macl) # end</pre>	Returns to privileged EXEC mode.
Step 5	show access-lists [<i>access-list-number</i> <i>access-list-name</i>] Example: <pre>Switch# show access-lists</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Classifying Traffic by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. The class map defines the criteria to use to match against a specific traffic flow to further classify it. Match statements can include criteria such as an ACL, IP precedence values, or DSCP values. The match criterion is defined with one match statement entered within the class-map configuration mode.



Note You can also create class maps during policy map creation by using the **class** policy-map configuration command.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
 - **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
 - **access-list** *access-list-number* {**deny** | **permit**} *protocol* *source* [*source-wildcard*] *destination* [*destination-wildcard*]
 - **ipv6 access-list** *access-list-name* {**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]
 - **mac access-list extended** *name* {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}
5. **end**

6. `show class-map`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>Use one of the following:</p> <ul style="list-style-type: none"> • <code>access-list access-list-number {deny permit} source [source-wildcard]</code> • <code>access-list access-list-number {deny permit} protocol source [source-wildcard] destination [destination-wildcard]</code> • <code>ipv6 access-list access-list-name {deny permit} protocol {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/ prefix-length any host destination-ipv6-address} [operator [port-number]] [dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]</code> • <code>mac access-list extended name {permit deny} {host src-MAC-addr mask any host dst-MAC-addr dst-MAC-addr mask} [type mask]</code> <p>Example:</p> <pre>Switch(config)# access-list 103 permit ip any any dscp 10</pre>	<p>Creates an IP standard or extended ACL, an IPv6 ACL for IP traffic, or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary.</p> <p>When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.</p>
Step 3	<p><code>class-map [match-all match-any] class-map-name</code></p> <p>Example:</p> <pre>Switch(config)# class-map class1</pre>	<p>Creates a class map, and enters class-map configuration mode.</p> <p>By default, no class maps are defined.</p> <ul style="list-style-type: none"> • (Optional) Use the match-all keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the match-any keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For <i>class-map-name</i>, specify the name of the class map.

	Command or Action	Purpose
		<p>If neither the match-all or match-any keyword is specified, the default is match-all.</p> <p>Note To delete an existing class map, use the no class-map [match-all match-any] class-map-name global configuration command.</p>
Step 4	<p>match {access-group <i>acl-index-or-name</i> ip dscp <i>dscp-list</i> ip precedence <i>ip-precedence-list</i>}</p> <p>Example:</p> <pre>Switch(config-cmap) # match ip dscp 10 11 12</pre>	<p>Defines the match criterion to classify traffic.</p> <p>By default, no match criterion is defined.</p> <p>Only one match criterion per class map is supported, and only one ACL per class map is supported.</p> <ul style="list-style-type: none"> • For access-group <i>acl-index-or-name</i>, specify the number or name of the ACL created in Step 2. • To filter IPv6 traffic with the match access-group command, create an IPv6 ACL, as described in Step 2. • For ip dscp <i>dscp-list</i>, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For ip precedence <i>ip-precedence-list</i>, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. <p>Note To remove a match criterion, use the no match {access-group <i>acl-index-or-name</i> ip dscp ip precedence} class-map configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-cmap) # end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show class-map</p> <p>Example:</p> <pre>Switch# show class-map</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Classifying Traffic by Using Class Maps and Filtering IPv6 Traffic

To apply the primary match criteria to only IPv4 traffic, use the **match protocol** command with the **ip** keyword. To apply the primary match criteria to only IPv6 traffic, use the **match protocol** command with the **ipv6** keyword.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {**match-all**} *class-map-name*
3. **match protocol** [*ip / ipv6*]
4. **match** {**ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}
5. **end**
6. **show class-map**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	class-map { match-all } <i>class-map-name</i> Example: <pre>Switch(config)# class-map cm-1</pre>	<p>Creates a class map, and enters class-map configuration mode.</p> <p>By default, no class maps are defined.</p> <p>When you use the match protocol command, only the match-all keyword is supported.</p> <ul style="list-style-type: none"> • For <i>class-map-name</i>, specify the name of the class map. <p>If neither the match-all or match-any keyword is specified, the default is match-all.</p> <p>Note To delete an existing class map, use the no class-map [match-all match-any] <i>class-map-name</i> global configuration command.</p>
Step 3	match protocol [<i>ip / ipv6</i>] Example: <pre>Switch(config-cmap)# match protocol ip</pre>	<p>(Optional) Specifies the IP protocol to which the class map applies:</p> <ul style="list-style-type: none"> • Use the argument <i>ip</i> to specify IPv4 traffic and <i>ipv6</i> to specify IPv6 traffic. • When you use the match protocol command, only the match-all keyword is supported for the class-map command.

	Command or Action	Purpose
Step 4	<p>match {ip dscp <i>dscp-list</i> ip precedence <i>ip-precedence-list</i>}</p> <p>Example:</p> <pre>Switch(config-cmap) # match ip dscp 10</pre>	<p>Defines the match criterion to classify traffic.</p> <p>By default, no match criterion is defined.</p> <ul style="list-style-type: none"> For ip dscp <i>dscp-list</i>, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. For ip precedence <i>ip-precedence-list</i>, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. <p>Note To remove a match criterion, use the no match {access-group <i>acl-index-or-name</i> ip dscp ip precedence} class-map configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-cmap) # end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show class-map</p> <p>Example:</p> <pre>Switch# show class-map</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps

You can configure a policy map on a physical port that specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; and specifying the traffic bandwidth limitations for each matched traffic class (policer) and the action to take when the traffic is out of profile (marking).

A policy map also has these characteristics:

- A policy map can contain multiple class statements, each with different match criteria and policers.
- A policy map can contain a predefined default traffic class explicitly placed at the end of the map.
- A separate policy-map class can exist for each type of traffic received through a port.

Follow these guidelines when configuring policy maps on physical ports:

- You can attach only one policy map per ingress port.
- If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp** *dscp1...dscp8* global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence** *new-precedence* policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want the egress DSCP value to be different than the ingress value, use the **set dscp new-dscp** policy-map class configuration command.
- If you enter or have used the **set ip dscp** command, the changes this command to **set dscp** in its configuration.
- You can use the **set ip precedence** or the **set precedence** policy-map class configuration command to change the packet IP precedence value. This setting appears as set ip precedence in the configuration.
- A policy-map and a port trust state can both run on a physical interface. The policy-map is applied before the port trust state.
- When you configure a default traffic class by using the **class class-default** policy-map configuration command, unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as the default traffic class (class-default).

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**match-all** | **match-any**] *class-map-name*
3. **policy-map** *policy-map-name*
4. **class** [*class-map-name* | **class-default**]
5. **trust** [**cos** | **dscp** | **ip-precedence**]
6. **set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*}
7. **police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]
8. **exit**
9. **exit**
10. **interface** *interface-id*
11. **service-policy input** *policy-map-name*
12. **end**
13. **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
14. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	class-map [match-all match-any] <i>class-map-name</i> Example:	Creates a class map, and enters class-map configuration mode.

	Command or Action	Purpose
	<pre>Switch(config)# class-map ipclass1</pre>	<p>By default, no class maps are defined.</p> <ul style="list-style-type: none"> • (Optional) Use the match-all keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the match-any keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For <i>class-map-name</i>, specify the name of the class map. <p>If neither the match-all or match-any keyword is specified, the default is match-all.</p>
Step 3	<p>policy-map <i>policy-map-name</i></p> <p>Example:</p> <pre>Switch(config-cmap)# policy-map flowit</pre>	<p>Creates a policy map by entering the policy map name, and enters policy-map configuration mode.</p> <p>By default, no policy maps are defined.</p> <p>The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.</p> <p>Note To delete an existing policy map, use the no policy-map <i>policy-map-name</i> global configuration command.</p>
Step 4	<p>class [<i>class-map-name</i> class-default]</p> <p>Example:</p> <pre>Switch(config-pmap)# class ipclass1</pre>	<p>Defines a traffic classification, and enters policy-map class configuration mode.</p> <p>By default, no policy map class-maps are defined.</p> <p>If a traffic class has already been defined by using the class-map global configuration command, specify its name for <i>class-map-name</i> in this command.</p> <p>A class-default traffic class is pre-defined and can be added to any policy. It is always placed at the end of a policy map. With an implied match any included in the class-default class, all packets that have not already matched the other traffic classes will match class-default.</p> <p>Note To delete an existing class map, use the no class <i>class-map-name</i> policy-map configuration command.</p>
Step 5	<p>trust [cos dscp ip-precedence]</p> <p>Example:</p>	<p>Configures the trust state, which QoS uses to generate a CoS-based or DSCP-based QoS label.</p>

	Command or Action	Purpose
	<pre>Switch(config-pmap-c)# trust dscp</pre>	<p>This command is mutually exclusive with the set command within the same policy map. If you enter the trust command, go to Step 6.</p> <p>By default, the port is not trusted. If no keyword is specified when the command is entered, the default is dscp.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • cos—QoS derives the DSCP value by using the received or default port CoS value and the CoS-to-DSCP map. • dscp—QoS derives the DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. • ip-precedence—QoS derives the DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. <p>Note To return to the untrusted state, use the no trust policy-map configuration command</p>
Step 6	<pre>set {dscp new-dscp ip precedence new-precedence}</pre> <p>Example:</p> <pre>Switch(config-pmap-c)# set dscp 45</pre>	<p>Classifies IP traffic by setting a new value in the packet.</p> <ul style="list-style-type: none"> • For dscp new-dscp, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. • For ip precedence new-precedence, enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7. <p>Note To remove an assigned DSCP or IP precedence value, use the no set {dscp new-dscp ip precedence new-precedence} policy-map configuration command.</p>
Step 7	<pre>police rate-bps burst-byte [exceed-action {drop policed-dscp-transmit}]</pre> <p>Example:</p> <pre>Switch(config-pmap-c)# police 100000</pre>	<p>Defines a policer for the classified traffic.</p> <p>By default, no policer is defined.</p> <ul style="list-style-type: none"> • For rate-bps, specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000.

	Command or Action	Purpose
	80000 drop	<ul style="list-style-type: none"> For <i>burst-byte</i>, specify the normal burst size in bytes. The range is 8000 to 1000000. (Optional) Specifies the action to take when the rates are exceeded. Use the exceed-action drop keywords to drop the packet. Use the exceed-action policed-dscp-transmit keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet. <p>Note To remove an existing policer, use the no police rate-bps burst-byte [exceed-action {drop policed-dscp-transmit}] policy-map configuration command.</p>
Step 8	exit Example: <pre>Switch(config-pmap-c)# exit</pre>	Returns to policy map configuration mode.
Step 9	exit Example: <pre>Switch(config-pmap)# exit</pre>	Returns to global configuration mode.
Step 10	interface interface-id Example: <pre>Switch(config)# interface gigabitethernet 2/0/1</pre>	<p>Specifies the port to attach to the policy map, and enters interface configuration mode.</p> <p>Valid interfaces include physical ports.</p>
Step 11	service-policy input policy-map-name Example: <pre>Switch(config-if)# service-policy input flowit</pre>	<p>Specifies the policy-map name, and applies it to an ingress port.</p> <p>Only one policy map per ingress port is supported.</p> <p>Note To remove the policy map and port association, use the no service-policy input policy-map-name interface configuration command.</p>
Step 12	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 13	show policy-map [<i>policy-map-name</i> [class <i>class-map-name</i>]] Example: Switch# show policy-map	Verifies your entries.
Step 14	copy running-config startup-config Example: Switch# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic by Using Aggregate Policers

By using an aggregate policer, you can create a policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or ports.

You can configure aggregate policers only in nonhierarchical policy maps on physical ports.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos aggregate-policer** *aggregate-policer-name* *rate-bps* *burst-byte* **exceed-action** {**drop** | **policed-dscp-transmit**}
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **policy-map** *policy-map-name*
5. **class** [*class-map-name* | **class-default**]
6. **police aggregate** *aggregate-policer-name*
7. **exit**
8. **interface** *interface-id*
9. **service-policy input** *policy-map-name*
10. **end**
11. **show mls qos aggregate-policer** [*aggregate-policer-name*]
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<p>mls qos aggregate-policer <i>aggregate-policer-name</i> <i>rate-bps</i> <i>burst-byte</i> exceed-action {drop policed-dscp-transmit}</p> <p>Example:</p> <pre>Switch(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action policed-dscp-transmit</pre>	<p>Defines the policer parameters that can be applied to multiple traffic classes within the same policy map.</p> <p>By default, no aggregate policer is defined.</p> <ul style="list-style-type: none"> For <i>aggregate-policer-name</i>, specify the name of the aggregate policer. For <i>rate-bps</i>, specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. For <i>burst-byte</i>, specify the normal burst size in bytes. The range is 8000 to 1000000. Specifies the action to take when the rates are exceeded. Use the exceed-action drop keywords to drop the packet. Use the exceed-action policed-dscp-transmit keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet.
Step 3	<p>class-map [match-all match-any] <i>class-map-name</i></p> <p>Example:</p> <pre>Switch(config)# class-map ipclass1</pre>	Creates a class map to classify traffic as necessary.
Step 4	<p>policy-map <i>policy-map-name</i></p> <p>Example:</p> <pre>Switch(config-cmap)# policy-map aggflow1</pre>	Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
Step 5	<p>class [<i>class-map-name</i> class-default]</p> <p>Example:</p> <pre>Switch(config-cmap-p)# class ipclass1</pre>	Defines a traffic classification, and enters policy-map class configuration mode.
Step 6	<p>police aggregate <i>aggregate-policer-name</i></p> <p>Example:</p> <pre>Switch(configure-cmap-p)# police aggregate transmit1</pre>	<p>Applies an aggregate policer to multiple classes in the same policy map.</p> <p>For <i>aggregate-policer-name</i>, enter the name specified in Step 2.</p> <p>To remove the specified aggregate policer from a policy map, use the no police aggregate <i>aggregate-policer-name</i> policy map configuration command. To delete an aggregate policer and its parameters, use the no mls qos aggregate-policer <i>aggregate-policer-name</i> global configuration command.</p>

	Command or Action	Purpose
Step 7	exit Example: Switch(configure-cmap-p)# exit	Returns to global configuration mode.
Step 8	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 2/0/1	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 9	service-policy input <i>policy-map-name</i> Example: Switch(config-if)# service-policy input aggflow1	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.
Step 10	end Example: Switch(configure-if)# end	Returns to privileged EXEC mode.
Step 11	show mls qos aggregate-policer [<i>aggregate-policer-name</i>] Example: Switch# show mls qos aggregate-policer transmit1	Verifies your entries.
Step 12	copy running-config startup-config Example: Switch# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring DSCP Maps

Configuring the CoS-to-DSCP Map

You use the CoS-to-DSCP map to map CoS values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Beginning in privileged EXEC mode, follow these steps to modify the CoS-to-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map cos-dscp *dscp1...dscp8***
3. **end**
4. **show mls qos maps cos-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	mls qos map cos-dscp <i>dscp1...dscp8</i> Example: Switch(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45	Modifies the CoS-to-DSCP map. For <i>dscp1...dscp8</i> , enter eight DSCP values that correspond to CoS values 0 to 7. Separate each DSCP value with a space. The DSCP range is 0 to 63. Note To return to the default map, use the no mls qos cos-dscp global configuration command.
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 4	show mls qos maps cos-dscp Example: Switch# show mls qos maps cos-dscp	Verifies your entries.
Step 5	copy running-config startup-config Example: Switch# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Beginning in privileged EXEC mode, follow these steps to modify the IP-precedence-to-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map ip-prec-dscp** *dscp1...dscp8*
3. **end**
4. **show mls qos maps ip-prec-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	mls qos map ip-prec-dscp <i>dscp1...dscp8</i> Example: <pre>Switch(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45</pre>	Modifies the IP-precedence-to-DSCP map. For <i>dscp1...dscp8</i> , enter eight DSCP values that correspond to the IP precedence values 0 to 7. Separate each DSCP value with a space. The DSCP range is 0 to 63. Note To return to the default map, use the no mls qos ip-prec-dscp global configuration command.
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	show mls qos maps ip-prec-dscp Example: <pre>Switch# show mls qos maps ip-prec-dscp</pre>	Verifies your entries.
Step 5	copy running-config startup-config Example: <pre>Switch# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the Policed-DSCP Map

You use the policed-DSCP map to mark down a DSCP value to a new value as the result of a policing and marking action.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the policed-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map policed-dscp *dscp-list to mark-down-dscp***
3. **end**
4. **show mls qos maps policed-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	mls qos map policed-dscp <i>dscp-list to mark-down-dscp</i> Example: <pre>Switch(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0</pre>	Modifies the policed-DSCP map. <ul style="list-style-type: none"> • For <i>dscp-list</i>, enter up to eight DSCP values separated by spaces. Then enter the to keyword. • For <i>mark-down-dscp</i>, enter the corresponding policed (marked down) DSCP value. <p>Note To return to the default map, use the no mls qos policed-dscp global configuration command.</p>
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	show mls qos maps policed-dscp Example: <pre>Switch(config)# show mls qos maps policed-dscp</pre>	Verifies your entries.

	Command or Action	Purpose
Step 5	copy running-config startup-config Example: <pre>Switch#</pre>	(Optional) Saves your entries in the configuration file.

Configuring the DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues. Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-CoS map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-cos *dscp-list* to *cos***
3. **end**
4. **show mls qos maps dscp-to-cos**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	mls qos map dscp-cos <i>dscp-list</i> to <i>cos</i> Example: <pre>Switch# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0</pre>	Modifies the DSCP-to-CoS map. <ul style="list-style-type: none"> • For <i>dscp-list</i>, enter up to eight DSCP values separated by spaces. Then enter the to keyword. • For <i>cos</i>, enter the CoS value to which the DSCP values correspond. The DSCP range is 0 to 63; the CoS range is 0 to 7. Note To return to the default map, use the no mls qos dscp-cos global configuration command.
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 4	show mls qos maps dscp-to-cos Example: <pre>Switch# show mls qos maps dscp-to-cos</pre>	Verifies your entries.
Step 5	copy running-config startup-config Example: <pre>Switch# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the DSCP-to-DSCP-Mutation Map

If two QoS domains have different DSCP definitions, use the DSCP-to-DSCP-mutation map to translate one set of DSCP values to match the definition of another domain. You apply the DSCP-to-DSCP-mutation map to the receiving port (ingress mutation) at the boundary of a QoS administrative domain.

With ingress mutation, the new DSCP value overwrites the one in the packet, and QoS applies the new value to the packet. The sends the packet out the port with the new DSCP value.

You can configure multiple DSCP-to-DSCP-mutation maps on an ingress port. The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-DSCP-mutation map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-mutation** *dscp-mutation-name in-dscp to out-dscp*
3. **interface** *interface-id*
4. **mls qos trust dscp**
5. **mls qos dscp-mutation** *dscp-mutation-name*
6. **end**
7. **show mls qos maps dscp-mutation**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<p>mls qos map dscp-mutation <i>dscp-mutation-name</i> <i>in-dscp</i> <i>to out-dscp</i></p> <p>Example:</p> <pre>Switch(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0</pre>	<p>Modifies the DSCP-to-DSCP-mutation map.</p> <ul style="list-style-type: none"> For <i>dscp-mutation-name</i>, enter the mutation map name. You can create more than one map by specifying a new name. For <i>in-dscp</i>, enter up to eight DSCP values separated by spaces. Then enter the to keyword. For <i>out-dscp</i>, enter a single DSCP value. <p>The DSCP range is 0 to 63.</p> <p>Note To return to the default map, use the no mls qos dscp-mutation <i>dscp-mutation-name</i> global configuration command.</p>
Step 3	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet1/0/1</pre>	<p>Specifies the port to which to attach the map, and enters interface configuration mode.</p> <p>Valid interfaces include physical ports.</p>
Step 4	<p>mls qos trust dscp</p> <p>Example:</p> <pre>Switch(config-if)# mls qos trust dscp</pre>	<p>Configures the ingress port as a DSCP-trusted port. By default, the port is not trusted.</p>
Step 5	<p>mls qos dscp-mutation <i>dscp-mutation-name</i></p> <p>Example:</p> <pre>Switch(config-if)# mls qos dscp-mutation mutation1</pre>	<p>Applies the map to the specified ingress DSCP-trusted port.</p> <p>For <i>dscp-mutation-name</i>, enter the mutation map name specified in Step 2.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show mls qos maps dscp-mutation</p> <p>Example:</p> <pre>Switch# show mls qos maps dscp-mutation</pre>	<p>Verifies your entries.</p>
Step 8	<p>copy running-config startup-config</p> <p>Example:</p>	<p>(Optional) Saves your entries in the configuration file.</p>

	Command or Action	Purpose
	Switch# <code>copy-running-config startup-config</code>	

Configuring Egress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the following modules. You need to make decisions about these characteristics:

- Which packets are mapped by DSCP or CoS value to each queue and threshold ID?
- What drop percentage thresholds apply to the queue-set (four egress queues per port), and how much reserved and maximum memory is needed for the traffic type?
- How much of the fixed buffer space is allocated to the queue-set?
- Does the bandwidth of the port need to be rate limited?
- How often should the egress queues be serviced and which technique (shaped, shared, or both) should be used?

Configuration Guidelines

Follow these guidelines when the expedite queue is enabled or the egress queues are serviced based on their SRR weights:

- If the egress expedite queue is enabled, it overrides the SRR shaped and shared weights for queue 1.
- If the egress expedite queue is disabled and the SRR shaped and shared weights are configured, the shaped mode overrides the shared mode for queue 1, and SRR services this queue in shaped mode.
- If the egress expedite queue is disabled and the SRR shaped weights are not configured, SRR services this queue in shared mode.

Allocating Buffer Space to and Setting WTD Thresholds for an Egress Queue-Set

You can guarantee the availability of buffers, set WTD thresholds, and configure the maximum allocation for a queue-set by using the **mls qos queue-set output *qset-id* threshold *queue-id* drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold** global configuration command.

Each threshold value is a percentage of the queue's allocated buffers, which you specify by using the **mls qos queue-set output *qset-id* buffers *allocation1* ... *allocation4*** global configuration command. The queues use WTD to support distinct drop percentages for different traffic classes.



Note The switch supports 4 egress queues by default, although there is an option to enable a total of 8 egress queues. Use the **mls qos srr-queue output queues 8** global configuration command to enable all 8 egress queues. Once 8 egress queues are enabled, you are able to configure thresholds, buffers, bandwidth share weights, and bandwidth shape weights for all 8 queues. The 8 egress queue configuration is only supported on a standalone switch.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to configure the memory allocation and to drop thresholds for a queue-set. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos srr-queue output queues 8**
3. **mls qos queue-set output *qset-id* buffers *allocation1* ... *allocation8***
4. **mls qos queue-set output *qset-id* threshold *queue-id* drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold**
5. **interface *interface-id***
6. **queue-set *qset-id***
7. **end**
8. **show mls qos interface [*interface-id*] buffers**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	mls qos srr-queue output queues 8 Example: <pre>Switch(config)# mls qos srr-queue output queues 8</pre>	(Optional) The switch supports 4 egress queues by default, although you can enable a total of 8 egress queues. Use the optional mls qos srr-queue output queues 8 command to enable the additional 4 egress queues. Once 8 queue support is enabled, you can then proceed to configure the additional 4 queues. Any existing egress queue configuration commands are then modified to support the additional queue parameters. Note The option to enable 8 queues is only available on a standalone switch.
Step 3	mls qos queue-set output <i>qset-id</i> buffers <i>allocation1</i> ... <i>allocation8</i> Example: <pre>Switch(config)# mls qos queue-set output 2 buffers 40 20 20 20 10 10 10 10</pre>	Allocates buffers to a queue set. By default, all allocation values are equally mapped among the four queues (25, 25, 25, 25). Each queue has 1/4 of the buffer space. When eight egress queues are configured, then by default 30 percent of the total buffer space is allocated

	Command or Action	Purpose
		<p>to queue 2 and 10 percent (each) to queues 1,3,4,5,6,7, and 8.</p> <p>If you enabled 8 egress queues as described in Step 2 above, then the following applies:</p> <ul style="list-style-type: none"> • For <i>qset-id</i>, enter the ID of the queue set. The range is 1 to 2. Each port belongs to a queue set, which defines all the characteristics of the four egress queues per port. • For <i>allocation1 ... allocation8</i>, specify eight percentages, one for each queue in the queue set. For <i>allocation1</i>, <i>allocation3</i>, and <i>allocation4</i> to <i>allocation8</i>, the range is 0 to 99. For <i>allocation2</i>, the range is 1 to 100 (including the CPU buffer). <p>Allocate buffers according to the importance of the traffic; for example, give a large percentage of the buffer to the queue with the highest-priority traffic.</p> <p>Note To return to the default setting, use the no mls qos queue-set output <i>qset-id</i> buffers global configuration command.</p>
<p>Step 4</p>	<p>mls qos queue-set output <i>qset-id</i> threshold <i>queue-id</i> <i>drop-threshold1</i> <i>drop-threshold2</i> <i>reserved-threshold</i> <i>maximum-threshold</i></p> <p>Example:</p> <pre>Switch(config)# mls qos queue-set output 2 threshold 2 40 60 100 200</pre>	<p>Configures the WTD thresholds, guarantee the availability of buffers, and configure the maximum memory allocation for the queue-set (four egress queues per port).</p> <p>By default, the WTD thresholds for queues 1, 3, and 4 are set to 100 percent. The thresholds for queue 2 are set to 200 percent. The reserved thresholds for queues 1, 2, 3, and 4 are set to 50 percent. The maximum thresholds for all queues are set to 400 percent by default.</p> <p>If you enabled 8 egress queues as described in Step 2 above, then the following applies:</p> <ul style="list-style-type: none"> • For <i>qset-id</i>, enter the ID of the queue-set specified in Step 2. The range is 1 to 2. • For <i>queue-id</i>, enter the specific queue in the queue set on which the command is performed. The queue-id range is 1-4 by default and 1-8 when 8 queues are enabled. • For <i>drop-threshold1</i> <i>drop-threshold2</i>, specify the two WTD thresholds expressed as a percentage of the queue's allocated memory. The range is 1 to 3200 percent. • For <i>reserved-threshold</i>, enter the amount of memory to be guaranteed (reserved) for the queue expressed

	Command or Action	Purpose
		<p>as a percentage of the allocated memory. The range is 1 to 100 percent.</p> <ul style="list-style-type: none"> For <i>maximum-threshold</i>, enable a queue in the full condition to obtain more buffers than are reserved for it. This is the maximum memory the queue can have before the packets are dropped if the common pool is not empty. The range is 1 to 3200 percent. <p>Note To return to the default WTD threshold percentages, use the no mls qos queue-set output <i>qset-id</i> threshold [<i>queue-id</i>] global configuration command.</p>
Step 5	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet1/0/1</pre>	Specifies the port of the outbound traffic, and enter interface configuration mode.
Step 6	<p>queue-set <i>qset-id</i></p> <p>Example:</p> <pre>Switch(config-id)# queue-set 2</pre>	<p>Maps the port to a queue-set.</p> <p>For <i>qset-id</i>, enter the ID of the queue-set specified in Step 2. The range is 1 to 2. The default is 1.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Switch(config-id)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show mls qos interface [<i>interface-id</i>] buffers</p> <p>Example:</p> <pre>Switch# show mls qos interface buffers</pre>	Verifies your entries.
Step 9	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy-running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default setting, use the no mls qos queue-set output <i>qset-id</i> buffers global configuration command. To return to the default WTD threshold percentages, use the no mls qos queue-set output <i>qset-id</i> threshold [<i>queue-id</i>] global configuration command.</p>

Mapping DSCP or CoS Values to an Egress Queue and to a Threshold ID

You can prioritize traffic by placing packets with particular DSCPs or costs of service into certain queues and adjusting the queue thresholds so that packets with lower priorities are dropped.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to map DSCP or CoS values to an egress queue and to a threshold ID. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
 - **mls qos srr-queue output dscp-map queue *queue-id* threshold *threshold-id* dscp1...dscp8**
 - **mls qos srr-queue output cos-map queue *queue-id* threshold *threshold-id* cos1...cos8**
3. **mls qos srr-queue output cos-map queue *queue-id* threshold *threshold-id* cos1...cos8**
4. **end**
5. **show mls qos maps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	Use one of the following: <ul style="list-style-type: none"> • mls qos srr-queue output dscp-map queue <i>queue-id</i> threshold <i>threshold-id</i> dscp1...dscp8 • mls qos srr-queue output cos-map queue <i>queue-id</i> threshold <i>threshold-id</i> cos1...cos8 Example: <pre>Switch(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11</pre>	Maps DSCP or CoS values to an egress queue and to a threshold ID. By default, DSCP values 0–15 are mapped to queue 2 and threshold 1. DSCP values 16–31 are mapped to queue 3 and threshold 1. DSCP values 32–39 and 48–63 are mapped to queue 4 and threshold 1. DSCP values 40–47 are mapped to queue 1 and threshold 1. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1. <ul style="list-style-type: none"> • For <i>queue-id</i>, the range is 1 to 4. <p>Note If you enabled 8 egress queues using the mls qos srr-queue output queues 8 global configuration command, then the <i>queue-id</i> range would be from 1 to 8.</p>

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>threshold-id</i>, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. For <i>dscp1...dscp8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 63. For <i>cos1...cos8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 7. <p>Note To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the no mls qos srr-queue output dscp-map or the no mls qos srr-queue output cos-map global configuration command.</p>
Step 3	<p>mls qos srr-queue output cos-map queue <i>queue-id</i> threshold <i>threshold-id</i> <i>cos1...cos8</i></p> <p>Example:</p> <pre>Switch(config)# mls qos srr-queue output cos-map queue 3 threshold 1 2 3</pre>	<p>Maps CoS values to an egress queue and to a threshold ID. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1.</p> <ul style="list-style-type: none"> For <i>queue-id</i>, the range is 1 to 4. For <i>threshold-id</i>, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. For <i>cos1...cos8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 7. <p>Note To return to the default CoS output queue threshold map, use the no mls qos srr-queue output cos-map global configuration command.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 5	<p>show mls qos maps</p> <p>Example:</p> <pre>Switch# show mls qos maps</pre>	<p>Verifies your entries.</p> <p>The DSCP output queue threshold map appears as a matrix. The d1 column specifies the most-significant digit of the DSCP number; the d2 row specifies the least-significant digit in the DSCP number. The intersection of the d1 and</p>

	Command or Action	Purpose
		<p>the d2 values provides the queue ID and threshold ID; for example, queue 2 and threshold 1 (02-01).</p> <p>The CoS output queue threshold map shows the CoS value in the top row and the corresponding queue ID and threshold ID in the second row; for example, queue 2 and threshold 2 (2-2).</p>
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy-running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the no mls qos srr-queue output dscp-map or the no mls qos srr-queue output cos-map global configuration command.</p>

Configuring SRR Shaped Weights on Egress Queues

You can specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the SRR scheduler sends packets from each queue.

You can configure the egress queues for shaped or shared weights, or both. Use shaping to smooth bursty traffic or to provide a smoother output over time.

Beginning in privileged EXEC mode, follow these steps to assign the shaped weights and to enable bandwidth shaping on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	Specifies the port of the outbound traffic, and enters interface configuration mode.

	Command or Action	Purpose
Step 3	<p>srr-queue bandwidth shape <i>weight1 weight2 weight3 weight4</i></p> <p>Example:</p> <pre>Switch(config-if)# srr-queue bandwidth shape 8 0 0 0</pre>	<p>Assigns SRR weights to the egress queues. By default, weight1 is set to 25; weight2, weight3, and weight4 are set to 0, and these queues are in shared mode.</p> <p>For <i>weight1 weight2 weight3 weight4</i>, enter the weights to control the percentage of the port that is shaped. The inverse ratio (1/weight) controls the shaping bandwidth for this queue. Separate each value with a space. The range is 0 to 65535.</p> <p>If you configure a weight of 0, the corresponding queue operates in shared mode. The weight specified with the srr-queue bandwidth shape command is ignored, and the weights specified with the srr-queue bandwidth share interface configuration command for a queue come into effect. When configuring queues in the same queue-set for both shaping and sharing, make sure that you configure the lowest number queue for shaping.</p> <p>The shaped mode overrides the shared mode.</p> <p>To return to the default setting, use the no srr-queue bandwidth shape interface configuration command.</p> <p>Note If you enabled 8 egress queues using the mls qos srr-queue output queues 8 global configuration command, then you would be able to assign SRR weights to a total of 8 queues.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show mls qos interface <i>interface-id queueing</i></p> <p>Example:</p> <pre>Switch# show mls qos interface interface-id queueing</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default setting, use the no srr-queue bandwidth shape interface configuration command.</p>

Configuring SRR Shared Weights on Egress Queues

In shared mode, the queues share the bandwidth among them according to the configured weights. The bandwidth is guaranteed at this level but not limited to it. For example, if a queue empties and does not require a share of the link, the remaining queues can expand into the unused bandwidth and share it among them. With sharing, the ratio of the weights controls the frequency of dequeuing; the absolute values are meaningless.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to assign the shared weights and to enable bandwidth sharing on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth share** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	Specifies the port of the outbound traffic, and enters interface configuration mode.
Step 3	srr-queue bandwidth share <i>weight1 weight2 weight3 weight4</i> Example: <pre>Switch(config-id)# srr-queue bandwidth share 1 2 3 4</pre>	<p>Assigns SRR weights to the egress queues. By default, all four weights are 25 (1/4 of the bandwidth is allocated to each queue).</p> <p>For <i>weight1 weight2 weight3 weight4</i>, enter the weights to control the ratio of the frequency in which the SRR scheduler sends packets. Separate each value with a space. The range is 1 to 255.</p> <p>To return to the default setting, use the no srr-queue bandwidth share interface configuration command.</p>

	Command or Action	Purpose
		<p>Note If you enabled 8 egress queues using the mls qos srr-queue output queues 8 global configuration command, then you would be able to assign SRR weights to a total of 8 queues.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config-id)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show mls qos interface <i>interface-id</i> queuing</p> <p>Example:</p> <pre>Switch# show mls qos interface interface_id queuing</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy-running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default setting, use the no srr-queue bandwidth share interface configuration command.</p>

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. SRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **interface *interface-id***
4. **priority-queue out**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	mls qos Example: Switch(config)# <code>mls qos</code>	Enables QoS on a switch.
Step 3	interface interface-id Example: Switch(config)# <code>interface gigabitethernet1/0/1</code>	Specifies the egress port, and enters interface configuration mode.
Step 4	priority-queue out Example: Switch(config-if)# <code>priority-queue out</code>	<p>Enables the egress expedite queue, which is disabled by default.</p> <p>When you configure this command, the SRR weight and queue size ratios are affected because there is one fewer queue participating in SRR. This means that <i>weight1</i> in the srr-queue bandwidth shape or the srr-queue bandwidth share command is ignored (not used in the ratio calculation).</p> <p>Note To disable the egress expedite queue, use the no priority-queue out interface configuration command.</p>
Step 5	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example:	<p>(Optional) Saves your entries in the configuration file.</p> <p>To disable the egress expedite queue, use the no priority-queue out interface configuration command.</p>

	Command or Action	Purpose
	<pre>Switch# copy running-config startup-config</pre>	

Limiting the Bandwidth on an Egress Interface

You can limit the bandwidth on an egress port. For example, if a customer pays only for a small percentage of a high-speed link, you can limit the bandwidth to that amount.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to limit the bandwidth on an egress port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth limit** *weight1*
4. **end**
5. **show mls qos interface** [*interface-id*] **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	Specifies the port to be rate-limited, and enters interface configuration mode.
Step 3	<p>srr-queue bandwidth limit <i>weight1</i></p> <p>Example:</p> <pre>Switch(config-if)# srr-queue bandwidth limit 80</pre>	<p>Specifies the percentage of the port speed to which the port should be limited. The range is 10 to 90.</p> <p>By default, the port is not rate-limited and is set to 100 percent.</p>

	Command or Action	Purpose
		Note To return to the default setting, use the no srr-queue bandwidth limit interface configuration command.
Step 4	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 5	show mls qos interface [interface-id] queueing Example: Switch# show mls qos interface interface_id queueing	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy-running-config startup-config	(Optional) Saves your entries in the configuration file. To return to the default setting, use the no srr-queue bandwidth limit interface configuration command.

Monitoring Standard QoS

Table 95: Commands for Monitoring Standard QoS on the Switch

Command	Description
show mls qos	Displays global QoS configuration information.
show mls qos aggregate-policer [aggregate-policer-name]	Displays the aggregate policer configuration.
show mls qos interface [interface-id] [buffers policers queueing statistics]	Displays QoS information at the port level, including the allocation, which ports have configured policers, the queueing strategy, and the ingress and egress statistics.
show mls qos maps [cos-dscp cos-output-q dscp-cos dscp-mutation dscp-mutation-name dscp-output-q ip-prec-dscp policed-dscp]	Displays QoS mapping information.
show mls qos queue-set [qset-id]	Displays QoS settings for the egress queues.
show running-config include rewrite	Displays the DSCP transparency setting.

Configuration Examples for QoS

Example: Configuring Port to the DSCP-Trusted State and Modifying the DSCP-to-DSCP-Mutation Map

This example shows how to configure a port to the DSCP-trusted state and to modify the DSCP-to-DSCP-mutation map (named *gi1/0/2-mutation*) so that incoming DSCP values 10 to 13 are mapped to DSCP 30:

```
Switch(config)# mls qos map dscp-mutation gigabitethernet1/0/2-mutation
10 11 12 13 to 30
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation gigabitethernet1/0/2-mutation
Switch(config-if)# end
```

Examples: Classifying Traffic by Using ACLs

This example shows how to allow access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses. Any host with a source address that does not match the access list statements is rejected.

```
Switch(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Switch(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Switch(config)# access-list 1 permit 36.0.0.0 0.0.0.255
! (Note: all other access implicitly denied)
```

This example shows how to create an ACL that permits IP traffic from any source to any destination that has the DSCP value set to 32:

```
Switch(config)# access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IP traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Switch(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

This example shows how to create an ACL that permits PIM traffic from any source to a destination group address of 224.0.0.2 with a DSCP set to 32:

```
Switch(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

This example shows how to create an ACL that permits IPv6 traffic from any source to any destination that has the DSCP value set to 32:

```
Switch(config)# ipv6 access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IPv6 traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Switch(config)# ipv6 access-list ipv6_Name_ACL permit ip host 10::1 host 10.1.1.2
precedence 5
```

This example shows how to create a Layer 2 MAC ACL with two permit statements. The first statement allows traffic from the host with MAC address 0001.0000.0001 to the host with MAC address 0002.0000.0001. The second statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 to the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
! (Note: all other access implicitly denied)
```

Examples: Classifying Traffic by Using Class Maps

This example shows how to configure the class map called *class1*. The *class1* has one match criterion, which is access list 103. It permits traffic from any host to any destination that matches a DSCP value of 10.

```
Switch(config)# access-list 103 permit ip any any dscp 10
Switch(config)# class-map class1
Switch(config-cmap)# match access-group 103
Switch(config-cmap)# end
Switch#
```

This example shows how to create a class map called *class2*, which matches incoming traffic with DSCP values of 10, 11, and 12.

```
Switch(config)# class-map class2
Switch(config-cmap)# match ip dscp 10 11 12
Switch(config-cmap)# end
Switch#
```

This example shows how to create a class map called *class3*, which matches incoming traffic with IP-precedence values of 5, 6, and 7:

```
Switch(config)# class-map class3
Switch(config-cmap)# match ip precedence 5 6 7
Switch(config-cmap)# end
Switch#
```

This example shows how to configure a class map to match IP DSCP and IPv6:

```
Switch(config)# Class-map cm-1
Switch(config-cmap)# match ip dscp 10
Switch(config-cmap)# match protocol ipv6
```



```
Switch(config-cmap)# exit
Switch(config)# Class-map cm-2
Switch(config-cmap)# match ip dscp 20
Switch(config-cmap)# match protocol ip
Switch(config-cmap)# exit
Switch(config)# Policy-map pml
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G1/0/1
Switch(config-if)# service-policy input pml
```

This example shows how to configure a class map that applies to both IPv4 and IPv6 traffic:

```
Switch(config)# ip access-list 101 permit ip any any
Switch(config)# ipv6 access-list ipv6-any permit ip any any
Switch(config)# Class-map cm-1
Switch(config-cmap)# match access-group 101
Switch(config-cmap)# exit
Switch(config)# class-map cm-2
Switch(config-cmap)# match access-group name ipv6-any
Switch(config-cmap)# exit
Switch(config)# Policy-map pml
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G0/1
Switch(config-if)# switch mode access
Switch(config-if)# service-policy input pml
```

Examples: Classifying, Policing, and Marking Traffic on Physical Ports Using Policy Maps

This example shows how to create a policy map and attach it to an ingress port. In the configuration, the IP standard ACL permits traffic from network 10.1.0.0. For traffic matching this classification, the DSCP value in the incoming packet is trusted. If the matched traffic exceeds an average traffic rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent:

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# policy-map flow1t
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police 1000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# service-policy input flow1t
```

This example shows how to create a Layer 2 MAC ACL with two permit statements and attach it to an ingress port. The first permit statement allows traffic from the host with MAC address 0001.0000.0001 destined for the host with MAC address 0002.0000.0001. The second permit statement allows only Ethernet XNS-IDP traffic from the host with MAC address 0001.0000.0002 destined for the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Switch(config-ext-mac)# exit
Switch(config)# mac access-list extended maclist2
Switch(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
Switch(config-ext-mac)# exit
Switch(config)# class-map macclass1
Switch(config-cmap)# match access-group maclist1
Switch(config-cmap)# exit
Switch(config)# policy-map macpolicy1
Switch(config-pmap)# class macclass1
Switch(config-pmap-c)# set dscp 63
Switch(config-pmap-c)# exit
Switch(config-pmap)# class macclass2 maclist2
Switch(config-pmap-c)# set dscp 45
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# service-policy input macpolicy1
```

This example shows how to create a class map that applies to both IPv4 and IPv6 traffic with the default class applied to unclassified traffic:

```
Switch(config)# ip access-list 101 permit ip any any
Switch(config)# ipv6 access-list ipv6-any permit ip any any
Switch(config)# class-map cm-1
Switch(config-cmap)# match access-group 101
Switch(config-cmap)# exit
Switch(config)# class-map cm-2
Switch(config-cmap)# match access-group name ipv6-any
Switch(config-cmap)# exit
Switch(config)# policy-map pml
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G0/1
Switch(config-if)# switch mode access
Switch(config-if)# service-policy input pml
```

Examples: Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps

This example shows how to create a hierarchical policy map:

```
Switch> enable
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# access-list 101 permit ip any any
Switch(config)# class-map cm-1
Switch(config-cmap)# match access 101
Switch(config-cmap)# exit
Switch(config)# exit
Switch#
Switch#
```

This example shows how to attach the new map to an SVI:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# class-map cm-interface-1
Switch(config-cmap)# match input gigabitethernet3/0/1 - gigabitethernet3/0/2
Switch(config-cmap)# exit
Switch(config)# policy-map port-plcmap
Switch(config-pmap)# class cm-interface-1
Switch(config-pmap-c)# police 900000 9000 exc policed-dscp-transmit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# policy-map vlan-plcmap
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 7
Switch(config-pmap-c)# service-policy port-plcmap-1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# service-policy port-plcmap-1
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-3
Switch(config-pmap-c)# service-policy port-plcmap-2
Switch(config-pmap-c)# set dscp 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-4
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# exit
Switch(config)# interface vlan 10
Switch(config-if)# service-policy input vlan-plcmap
Switch(config-if)# exit
Switch(config)# exit
Switch#
```

This example shows that when a child-level policy map is attached below a class, an action must be specified for the class:

```
Switch(config)# policy-map vlan-plcmap
Switch(config-pmap)# class cm-5
Switch(config-pmap-c)# set dscp 7
```

```
Switch(config-pmap-c) # service-policy port-plcmap-1
```

This example shows how to configure a class map to match IP DSCP and IPv6:

```
Switch(config) # class-map cm-1
Switch(config-cmap) # match ip dscp 10
Switch(config-cmap) # match protocol ipv6
Switch(config-cmap) # exit
Switch(config) # class-map cm-2
Switch(config-cmap) # match ip dscp 20
Switch(config-cmap) # match protocol ip
Switch(config-cmap) # exit
Switch(config) # policy-map pm1
Switch(config-pmap) # class cm-1
Switch(config-pmap-c) # set dscp 4
Switch(config-pmap-c) # exit
Switch(config-pmap) # class cm-2
Switch(config-pmap-c) # set dscp 6
Switch(config-pmap-c) # exit
Switch(config-pmap) # exit
Switch(config) # interface G1/0/1
Switch(config-if) # service-policy input pm1
```

This example shows how to configure default traffic class to a policy map:

```
Switch# configure terminal
Switch(config) # class-map cm-3
Switch(config-cmap) # match ip dscp 30
Switch(config-cmap) # match protocol ipv6
Switch(config-cmap) # exit
Switch(config) # class-map cm-4
Switch(config-cmap) # match ip dscp 40
Switch(config-cmap) # match protocol ip
Switch(config-cmap) # exit
Switch(config) # policy-map pm3
Switch(config-pmap) # class class-default
Switch(config-pmap) # set dscp 10
Switch(config-pmap-c) # exit
Switch(config-pmap) # class cm-3
Switch(config-pmap-c) set dscp 4
Switch(config-pmap-c) # exit
Switch(config-pmap) # class cm-4
Switch(config-pmap-c) # trust cos
Switch(config-pmap-c) # exit
Switch(config-pmap) # exit
```

This example shows how the default traffic class is automatically placed at the end of policy-map pm3 even though class-default was configured first:

```
Switch# show policy-map pm3
Policy Map pm3
  Class cm-3
    set dscp 4
  Class cm-4
    trust cos
  Class class-default
    police 8000 80000 exceed-action drop
```

```
Switch#
```

Examples: Classifying, Policing, and Marking Traffic by Using Aggregate Policers

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. In the configuration, the IP ACLs permit traffic from network 10.1.0.0 and from host 11.3.1.1. For traffic coming from network 10.1.0.0, the DSCP in the incoming packets is trusted. For traffic coming from host 11.3.1.1, the DSCP in the packet is changed to 56. The traffic rate from the 10.1.0.0 network and from host 11.3.1.1 is policed. If the traffic exceeds an average rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent. The policy map is attached to an ingress port.

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# access-list 2 permit 11.3.1.1
Switch(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action
policed-dscp-transmit
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map ipclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map aggflow1
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class ipclass2
Switch(config-pmap-c)# set dscp 56
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# service-policy input aggflow1
Switch(config-if)# exit
```

Examples: Configuring DSCP Maps

This example shows how to modify and display the CoS-to-DSCP map:

```
Switch(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps cos-dscp

Cos-dscp map:
  cos:   0  1  2  3  4  5  6  7
-----
  dscp:  10 15 20 25 30 35 40 45
```

This example shows how to modify and display the IP-precedence-to-DSCP map:

```
Switch(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps ip-prec-dscp

IpPrecedence-dscp map:
  ipprec:  0  1  2  3  4  5  6  7
-----
  dscp:   10 15 20 25 30 35 40 45
```

This example shows how to map DSCP 50 to 57 to a marked-down DSCP value of 0:

```
Switch(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0
Switch(config)# end
Switch# show mls qos maps policed-dscp
Policed-dscp map:
  d1 :  d2 0  1  2  3  4  5  6  7  8  9
-----
  0 :   00 01 02 03 04 05 06 07 08 09
  1 :   10 11 12 13 14 15 16 17 18 19
  2 :   20 21 22 23 24 25 26 27 28 29
  3 :   30 31 32 33 34 35 36 37 38 39
  4 :   40 41 42 43 44 45 46 47 48 49
  5 :   00 00 00 00 00 00 00 00 58 59
  6 :   60 61 62 63
```



Note In this policed-DSCP map, the marked-down DSCP values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the marked-down value. For example, an original DSCP value of 53 corresponds to a marked-down DSCP value of 0.

This example shows how to map DSCP values 0, 8, 16, 24, 32, 40, 48, and 50 to CoS value 0 and to display the map:

```
Switch(config)# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0
Switch(config)# end
Switch# show mls qos maps dscp-cos
Dscp-cos map:
  d1 :  d2 0  1  2  3  4  5  6  7  8  9
-----
  0 :   00 00 00 00 00 00 00 00 00 01
  1 :   01 01 01 01 01 01 00 02 02 02
  2 :   02 02 02 02 00 03 03 03 03 03
  3 :   03 03 00 04 04 04 04 04 04 04
  4 :   00 05 05 05 05 05 05 05 00 06
  5 :   00 06 06 06 06 06 07 07 07 07
  6 :   07 07 07 07
```



Note In the above DSCP-to-CoS map, the CoS values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the DSCP; the d2 row specifies the least-significant digit of the DSCP. The intersection of the d1 and d2 values provides the CoS value. For example, in the DSCP-to-CoS map, a DSCP value of 08 corresponds to a CoS value of 0.

This example shows how to define the DSCP-to-DSCP-mutation map. All the entries that are not explicitly configured are not modified (remains as specified in the null map):

```
Switch(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0
Switch(config)# mls qos map dscp-mutation mutation1 8 9 10 11 12 13 to 10
Switch(config)# mls qos map dscp-mutation mutation1 20 21 22 to 20
Switch(config)# mls qos map dscp-mutation mutation1 30 31 32 33 34 to 30
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation mutation1
Switch(config-if)# end
Switch# show mls qos maps dscp-mutation mutation1
Dscp-dscp mutation map:
  mutation1:
    d1 :  d2 0  1  2  3  4  5  6  7  8  9
    -----
    0 :   00 00 00 00 00 00 00 00 10 10
    1 :   10 10 10 10 14 15 16 17 18 19
    2 :   20 20 20 23 24 25 26 27 28 29
    3 :   30 30 30 30 30 35 36 37 38 39
    4 :   40 41 42 43 44 45 46 47 48 49
    5 :   50 51 52 53 54 55 56 57 58 59
    6 :   60 61 62 63
```



Note In the above DSCP-to-DSCP-mutation map, the mutated values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the mutated value. For example, a DSCP value of 12 corresponds to a mutated value of 10.

Examples: Configuring Egress Queue Characteristics

This example shows how to map DSCP values 10 and 11 to egress queue 1 and to threshold 2:

```
Switch(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11
```

This example shows how to configure bandwidth shaping on queue 1. Because the weight ratios for queues 2, 3, and 4 are set to 0, these queues operate in shared mode. The bandwidth weight for queue 1 is 1/8, which is 12.5 percent:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth shape 8 0 0 0
```

This example shows how to configure the weight ratio of the SRR scheduler running on an egress port. Four queues are used, and the bandwidth ratio allocated for each queue in shared mode is $1/(1+2+3+4)$, $2/(1+2+3+4)$, $3/(1+2+3+4)$, and $4/(1+2+3+4)$, which is 10 percent, 20 percent, 30 percent, and 40 percent for queues 1, 2, 3, and 4. This means that queue 4 has four times the bandwidth of queue 1, twice the bandwidth of queue 2, and one-and-a-third times the bandwidth of queue 3.

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth share 1 2 3 4
```

This example shows how to enable the egress expedite queue when the SRR weights are configured. The egress expedite queue overrides the configured SRR weights.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# srr-queue bandwidth shape 25 0 0 0
Switch(config-if)# srr-queue bandwidth share 30 20 25 25
Switch(config-if)# priority-queue out
Switch(config-if)# end
```

This example shows how to limit the bandwidth on a port to 80 percent:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth limit 80
```

When you configure this command to 80 percent, the port is idle 20 percent of the time. The line rate drops to 80 percent of the connected speed, which is 800 Mb/s. These values are not exact because the hardware adjusts the line rate in increments of six.

Where to Go Next

Review the auto-QoS documentation to see if you can use these automated capabilities for your QoS configuration.



CHAPTER 46

Configuring Auto-QoS

- [Finding Feature Information, on page 921](#)
- [Prerequisites for Auto-QoS, on page 921](#)
- [Information about Configuring Auto-QoS, on page 922](#)
- [How to Configure Auto-QoS, on page 926](#)
- [Monitoring Auto-QoS, on page 928](#)
- [Configuration Examples for Auto-QoS, on page 929](#)
- [Where to Go Next for Auto-QoS, on page 937](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Auto-QoS

Before configuring standard QoS or auto-QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. Is the traffic bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

Information about Configuring Auto-QoS

Auto-QoS Overview

You can use the auto-QoS feature to simplify the deployment of QoS features. Auto-QoS determines the network design and enables QoS configurations so that the switch can prioritize different traffic flows. It uses the egress queues instead of using the default (disabled) QoS behavior. The switch offers best-effort service to each packet, regardless of the packet contents or size, and sends it from a single queue.

When you enable auto-QoS, it automatically classifies traffic based on the traffic type and ingress packet label. The switch uses the classification results to choose the appropriate egress queue.

You can use auto-QoS commands to identify ports connected to the following Cisco devices:

- Cisco IP Phones
- Devices running the Cisco SoftPhone application
- Cisco TelePresence
- Cisco IP Camera
- Cisco digital media player

You also use the auto-QoS commands to identify ports that receive trusted traffic through an uplink. Auto-QoS then performs these functions:

- Detects the presence or absence of auto-QoS devices through conditional trusted interfaces.
- Configures QoS classification
- Configures egress queues

Generated Auto-QoS Configuration

By default, auto-QoS is disabled on all ports. Packets are not modified--the CoS, DSCP and IP precedence values in the packet are not changed.

When you enable the auto-QoS feature on the first port of the interface:

- Ingress packet label is used to categorize traffic, to assign packet labels, and to configure the ingress and egress queues.
- QoS is globally enabled (**mls qos** global configuration command), and other global configuration commands are automatically generated. (See [Examples: Global Auto-QoS Configuration, on page 929](#)).
- Switch enables the trusted boundary feature and uses the Cisco Discovery Protocol (CDP) to detect the presence of a supported device.
- Policing is used to determine whether a packet is in or out of profile and specifies the action on the packet.

VoIP Device Specifics

The following activities occur when you issue these auto-QoS commands on a port:

- When you enter the **auto qos voip cisco-phone** command on a port at the network edge connected to a Cisco IP Phone, the switch enables the trusted boundary feature. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0. When there is no Cisco IP Phone, the ingress classification is set to not trust the QoS label in the packet. The policing is applied to the traffic matching the policy-map classification before the switch enables the trust boundary feature.
- When you enter the **auto qos voip cisco-softphone** interface configuration command on a port at the network edge that is connected to a device running the Cisco SoftPhone, the switch uses policing to determine whether a packet is in or out of profile and to specify the action on the packet. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0.
- When you enter the **auto qos voip trust** interface configuration command on a port connected to the network interior, the switch trusts the CoS value for nonrouted ports or the DSCP value for routed ports in ingress packets (the assumption is that traffic has already been classified by other edge devices).

Table 96: Traffic Types, Packet Labels, and Queues

	VoIP Data Traffic	VoIP Control Traffic	Routing Protocol Traffic	STP BPDU Traffic	Real-Time Video Traffic	All Other Traffic
DSCP value	46	24, 26	48	56	34	–
CoS value	5	3	6	7	3	–
CoS-to-Egress queue map	4, 5 (queue 1)	2, 3, 6, 7 (queue 2)			0 (queue 3)	2 (queue 3)

The switch configures egress queues on the port according to the settings in the following table. This table shows the generated auto-QoS configuration for the egress queues.

Table 97: Auto-QoS Configuration for the Egress Queues

Egress Queue	Egress Queue	Queue Number	Queue Weight (Bandwidth)	Queue Size (Giga Ports)
Priority	1	4, 5	up to 100 percent	25 p
SRR shared	2	2, 3, 6, 7	10 percent	25 p
SRR shared	3	0	60 percent	25 p
SRR shared	4	1	20 percent	25 p

- When you enable auto-QoS by using the **auto qos voip cisco-phone**, the **auto qos voip cisco-softphone**, or the **auto qos voip trust** interface configuration command, the switch automatically generates a QoS

configuration based on the traffic type and ingress packet label and applies the commands listed in [Examples: Global Auto-QoS Configuration, on page 929](#) to the port.

Enhanced Auto-QoS for Video, Trust, and Classification

Auto-QoS is enhanced to support video. Automatic configurations are generated that classify and trust traffic from Cisco TelePresence systems and Cisco IP cameras.

Auto-QoS Configuration Migration

Auto-QoS configuration migration from legacy auto-QoS to enhanced auto-QoS occurs when:

- A switch is booted with a 12.2(55)SE image and QoS is not enabled.
Any video or voice trust configuration on the interface automatically generates enhanced auto-QoS commands.
- A switch is enabled with QoS, these guidelines take effect:
 - If you configure the interface for conditional trust on a voice device, only the legacy auto-QoS VoIP configuration is generated.
 - If you configure the interface for conditional trust on a video device, the enhanced auto-QoS configuration is generated.
 - If you configure the interface with classification or conditional trust based on the new interface auto-QoS commands, enhanced auto-QoS configuration is generated.
- Auto-QoS migration happens after a new device is connected when the **auto qos srnd4** global configuration command is enabled.



Note If an interface previously configured with legacy auto-QoS migrates to enhanced auto-QoS, voice commands and configuration are updated to match the new global QoS commands.

Auto-QoS configuration migration from enhanced auto-QoS to legacy auto-QoS can occur only when you disable all existing auto-QoS configurations from the interface.

Auto-QoS Configuration Guidelines

Before configuring auto-QoS, you should be aware of this information:

- After auto-QoS is enabled, do not modify a policy map that includes *AutoQoS* in its name. If you need to modify the policy map, make a copy of it, and change the copied policy map. To use this new policy map instead of the generated one, remove the generated policy map from the interface, and apply the new policy map to the interface.
- To take advantage of the auto-QoS defaults, you should enable auto-QoS before you configure other QoS commands. If necessary, you can fine-tune the QoS configuration, but we recommend that you do so only after the auto-QoS configuration is completed.
- You can enable auto-QoS on static, dynamic-access, voice VLAN access, and trunk ports.

- By default, the CDP is enabled on all ports. For auto-QoS to function properly, do not disable CDP.

Auto-QoS VoIP Considerations

Before configuring auto-QoS for VoIP, you should be aware of this information:

- Auto-QoS configures the switch for VoIP with Cisco IP Phones on nonrouted and routed ports. Auto-QoS also configures the switch for VoIP with devices running the Cisco SoftPhone application.



Note When a device running Cisco SoftPhone is connected to a nonrouted or routed port, the switch supports only one Cisco SoftPhone application per port.

- When enabling auto-QoS with a Cisco IP Phone on a routed port, you must assign a static IP address to the IP phone.
- This release supports only Cisco IP SoftPhone Version 1.3(3) or later.
- Connected devices must use Cisco Call Manager Version 4 or later.

Auto-QoS Enhanced Considerations

Auto-QoS is enhanced to support video. Automatic configurations are generated that classify and trust traffic from Cisco TelePresence systems and Cisco IP cameras.

Before configuring auto-QoS enhanced, you should be aware of this information:

- The **auto qos srnd4** global configuration command is generated as a result of enhanced auto-QoS configuration.

Effects of Auto-QoS on Running Configuration

When auto-QoS is enabled, the **auto qos** interface configuration commands and the generated global configuration are added to the running configuration.

The switch applies the auto-QoS-generated commands as if the commands were entered from the CLI. An existing user configuration can cause the application of the generated commands to fail or to be overridden by the generated commands. These actions may occur without warning. If all the generated commands are successfully applied, any user-entered configuration that was not overridden remains in the running configuration. Any user-entered configuration that was overridden can be retrieved by reloading the switch without saving the current configuration to memory. If the generated commands are not applied, the previous running configuration is restored.

How to Configure Auto-QoS

Configuring Auto-QoS

Enabling Auto-QoS

For optimum QoS performance, enable auto-QoS on all the devices in your network.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**}
 - **auto qos video** {**cts** | **ip-camera** | **media-player**}
 - **auto qos classify** [**police**]
 - **auto qos trust** {**cos** | **dscp**}
4. **exit**
5. **interface** *interface-id*
6. **auto qos trust**
7. **end**
8. **show auto qos interface** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 3/0/1	Specifies the port that is connected to a video device or the uplink port that is connected to another trusted switch or router in the network interior, and enters interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • auto qos voip {cisco-phone cisco-softphone trust} • auto qos video {cts ip-camera media-player} • auto qos classify [police] • auto qos trust {cos dscp} 	Enables auto-QoS for VoIP. <ul style="list-style-type: none"> • cisco-phone—If the port is connected to a Cisco IP Phone, the QoS labels of incoming packets are trusted only when the telephone is detected.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(config-if)# auto qos trust dscp</pre>	<ul style="list-style-type: none"> • cisco-softphone—The port is connected to device running the Cisco SoftPhone feature. • trust—The uplink port is connected to a trusted switch or router, and the VoIP traffic classification in the ingress packet is trusted. <p>Enables auto-QoS for a video device.</p> <ul style="list-style-type: none"> • cts—A port connected to a Cisco Telepresence system. • ip-camera—A port connected to a Cisco video surveillance camera. • media-player—A port connected to a CDP-capable Cisco digital media player. <p>QoS labels of incoming packets are trusted only when the system is detected.</p> <p>Enables auto-QoS for classification.</p> <ul style="list-style-type: none"> • police—Policing is set up by defining the QoS policy maps and applying them to ports (port-based QoS). <p>Enables auto-QoS for trusted interfaces.</p> <ul style="list-style-type: none"> • cos—Class of service. • dscp—Differentiated Services Code Point. • <cr>—Trust interface.
Step 4	<p>exit</p> <p>Example:</p> <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 5	<p>interface interface-id</p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet 2/0/1</pre>	Specifies the switch port identified as connected to a trusted switch or router, and enters interface configuration mode.
Step 6	<p>auto qos trust</p> <p>Example:</p> <pre>Switch(config-if)# auto qos trust</pre>	Enables auto-QoS on the port, and specifies that the port is connected to a trusted router or switch.

	Command or Action	Purpose
Step 7	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 8	show auto qos interface <i>interface-id</i> Example: <pre>Switch# show auto qos interface gigabitethernet 2/0/1</pre>	Verifies your entries. This command displays the auto-QoS command on the interface on which auto-QoS was enabled. You can use the show running-config privileged EXEC command to display the auto-QoS configuration and the user modifications.

Troubleshooting Auto-QoS

To troubleshoot auto-QoS, use the **debug auto qos** privileged EXEC command. For more information, see the **debug auto qos** command in the command reference for this release.

To disable auto-QoS on a port, use the **no** form of the **auto qos** command interface configuration command, such as **no auto qos voip**. Only the auto-QoS-generated interface configuration commands for this port are removed. If this is the last port on which auto-QoS is enabled and you enter the **no auto qos voip** command, auto-QoS is considered disabled even though the auto-QoS-generated global configuration commands remain (to avoid disrupting traffic on other ports affected by the global configuration).

Monitoring Auto-QoS

Table 98: Commands for Monitoring Auto-QoS

Command	Description
show auto qos [interface [<i>interface-type</i>]]	Displays the initial auto-QoS configuration. You can compare the show auto qos and the show running-config commands to identify the user-defined QoS settings.
show mls qos aggregate policer <i>policer_name</i>	Displays information about the QoS aggregate policers affected by auto-QoS.
show mls qos interface [<i>interface-type</i> buffers policers queueing statistics]	Displays information about the QoS interface configuration affected by auto-QoS.
show mls qos maps [cos-dscp cos-output-q dscp-cos dscp-mutation dscp-output-q ip-prec-dscp policed-dscp]	Displays information about the QoS maps configured by auto-QoS.
show mls qos queue-set <i>queue-set ID</i>	Displays information about the QoS queue-sets configured by auto-QoS.
show mls qos stack-port buffers	Displays information about the QoS stack ports affected by auto-QoS.

Command	Description
<code>show mls qos stack-qset</code>	Displays information about the QoS stack affected by auto-QoS.
<code>show running-config</code>	Displays information about the QoS configuration. You can compare the <code>show auto qos</code> and <code>show running-config</code> to identify the user-defined QoS settings.

Configuration Examples for Auto-QoS

Examples: Global Auto-QoS Configuration

The following table describes the automatically generated commands for auto-QoS and enhanced auto-QoS by the switch.

Table 99: Generated Auto-QoS Configuration

Description	Automatically Generated Command {voip}	Enhanced
The switch automatically enables standard QoS and configures the CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value).	<pre>Switch(config)# mls qos Switch(config)# mls qos map cos-dscp 0 8 16 26 32 46 48 56</pre>	<pre>Switch(c Switch(c 0 8 16 2</pre>
The switch automatically maps CoS values to an egress queue and to a threshold ID.	<pre>Switch(config)# no mls qos srr-queue output cos-map Switch(config)# mls qos srr-queue output cos-map queue 1 threshold 3 5 Switch(config)# mls qos srr-queue output cos-map queue 2 threshold 3 3 6 7 Switch(config)# mls qos srr-queue output cos-map queue 3 threshold 3 2 4 Switch(config)# mls qos srr-queue output cos-map queue 4 threshold 2 1 Switch(config)# mls qos srr-queue output cos-map queue 4 threshold 3 0</pre>	<pre>Switch(c output c Switch(c output c Switch(c output c Switch(c output c Switch(c output c Switch(c output c Switch(c output c</pre>

Description	Automatically Generated Command {voip}	Enhanced Au
<p>The switch automatically maps DSCP values to an egress queue and to a threshold ID.</p>	<pre>Switch(config)# no mls qos srr-queue output dscp-map Switch(config)# mls qos srr-queue output dscp-map queue 1 threshold 3 40 41 42 43 44 45 46 47 Switch(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 24 25 26 27 28 29 30 31 Switch(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 48 49 50 51 52 53 54 55 Switch(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 56 57 58 59 60 61 62 63 Switch(config)# mls qos srr-queue output dscp-map queue 3 threshold 3 16 17 18 19 20 21 22 23 Switch(config)# mls qos srr-queue output dscp-map queue 3 threshold 3 32 33 34 35 36 37 38 39 Switch(config)# mls qos srr-queue output dscp-map queue 4 threshold 1 8 Switch(config)# mls qos srr-queue output dscp-map queue 4 threshold 2 9 10 11 12 13 14 15 Switch(config)# mls qos srr-queue output dscp-map queue 4 threshold 3 0 1 2 3 4 5 6 7</pre>	<pre>Switch(confi output dscp Switch(confi output dscp 33 40 41 42 Switch(confi output dscp 17 18 19 20 Switch(confi output dscp 27 28 29 30 Switch(confi output dscp Switch(confi output dscp 49 50 51 52 Switch(confi output dscp 58 59 60 61 Switch(confi output dscp 1 2 3 4 5 6 Switch(confi output dscp 9 11 13 15 Switch(confi output dscp 12 14</pre>

Description	Automatically Generated Command {voip}	Enhanced
The switch automatically configures the egress queue buffer sizes. It configures the bandwidth and the SRR mode (shaped or shared) on the egress queues mapped to the port.	<pre>Switch(config)# mls qos queue-set output 1 threshold 1 138 138 92 138 Switch(config)# mls qos queue-set output 1 threshold 2 138 138 92 400 Switch(config)# mls qos queue-set output 1 threshold 3 36 77 100 318 Switch(config)# mls qos queue-set output 1 threshold 4 20 50 67 400 Switch(config)# mls qos queue-set output 2 threshold 1 149 149 100 149 Switch(config)# mls qos queue-set output 2 threshold 2 118 118 100 235 Switch(config)# mls qos queue-set output 2 threshold 3 41 68 100 272 Switch(config)# mls qos queue-set output 2 threshold 4 42 72 100 242 Switch(config)# mls qos queue-set output 1 buffers 10 10 26 54 Switch(config)# mls qos queue-set output 2 buffers 16 6 17 61 Switch(config-if)# priority-queue out Switch(config-if)# srr-queue bandwidth share 10 10 60 20</pre>	<pre>Switch(c output 1 Switch(c output 1 Switch(c output 1 Switch(c output 1 Switch(c output 1 Switch(c output 1</pre>

Examples: Auto-QoS Generated Configuration for VoIP Devices

The following table describes the automatically generated commands for auto-QoS for VoIP devices by the switch.

Table 100: Generated Auto-QoS Configuration for VoIP Devices

Description	Automatically Generated Command (VoIP)
The switch automatically enables standard QoS and configures the CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value).	<pre>Switch(config)# mls qos Switch(config)# mls qos map cos-dscp 0 8 16 26 32 46 48 56</pre>
The switch automatically maps CoS values to an egress queue and to a threshold ID.	<pre>Switch(config)# no mls qos srr-queue output cos-map Switch(config)# mls qos srr-queue output cos-map queue 1 threshold 3 5 Switch(config)# mls qos srr-queue output cos-map queue 2 threshold 3 3 6 7 Switch(config)# mls qos srr-queue output cos-map queue 3 threshold 3 2 4 Switch(config)# mls qos srr-queue output cos-map queue 4 threshold 2 1 Switch(config)# mls qos srr-queue output cos-map queue 4 threshold 3 0</pre>

Description	Automatically Generated Command (VoIP)
The switch automatically maps DSCP values to an egress queue and to a threshold ID.	<pre>Switch(config)# no mls qos srr-queue output dscp-map Switch(config)# mls qos srr-queue output dscp-map queue 1 threshold 3 40 41 42 43 44 45 46 47 Switch(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 24 25 26 27 28 29 30 31 Switch(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 48 49 50 51 52 53 54 55 Switch(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 56 57 58 59 60 61 62 63 Switch(config)# mls qos srr-queue output dscp-map queue 3 threshold 3 16 17 18 19 20 21 22 23 Switch(config)# mls qos srr-queue output dscp-map queue 3 threshold 3 32 33 34 35 36 37 38 39 Switch(config)# mls qos srr-queue output dscp-map queue 4 threshold 1 8 Switch(config)# mls qos srr-queue output dscp-map queue 4 threshold 2 9 10 11 12 13 14 15 Switch(config)# mls qos srr-queue output dscp-map queue 4 threshold 3 0 1 2 3 4 5 6 7</pre>
The switch automatically configures the egress queue buffer sizes. It configures the bandwidth and the SRR mode (shaped or shared) on the egress queues mapped to the port.	<pre>SwitchSwitch(config)# mls qos queue-set output 1 threshold 1 138 138 92 138 Switch(config)# mls qos queue-set output 1 threshold 2 138 138 92 400 Switch(config)# mls qos queue-set output 1 threshold 3 36 77 100 318 Switch(config)# mls qos queue-set output 1 threshold 4 20 50 67 400 Switch(config)# mls qos queue-set output 2 threshold 1 149 149 100 149 Switch(config)# mls qos queue-set output 2 threshold 2 118 118 100 235 Switch(config)# mls qos queue-set output 2 threshold 3 41 68 100 272 Switch(config)# mls qos queue-set output 2 threshold 4 42 72 100 242 Switch(config)# mls qos queue-set output 1 buffers 10 10 26 54 Switch(config)# mls qos queue-set output 2 buffers 16 6 17 61 Switch(config-if)# priority-que out Switch(config-if)# srr-queue bandwidth share 10 10 60 20</pre>

If you entered the **auto qos voip cisco-phone** command, the switch automatically enables the trusted boundary feature, which uses the CDP to detect the presence or absence of a Cisco IP Phone (as shown below).

```
Switch(config-if)# mls qos trust device cisco-phone
```

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps (as shown below).

```
Switch(config)# mls qos map policed-dscp 24 26 46 to 0
```

```
Switch(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AutoQoS-VoIP-Control-Trust
Switch(config-cmap)# match ip dscp cs3 af31
Switch(config)# policy-map AutoQoS-Police-SoftPhone
Switch(config-pmap)# class AutoQoS-VoIP-RTP-Trust
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AutoQoS-VoIP-Control-Trust
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled (as shown below).

```
Switch(config-if)# service-policy input AutoQoS-Police-SoftPhone
```

Examples: Auto-QoS Generated Configuration for VoIP Devices

If you entered the **auto qos voip cisco-phone** command, the switch automatically enables the trusted boundary feature, which uses the CDP to detect the presence or absence of a Cisco IP Phone.

```
Switch(config-if)# mls qos trust device cisco-phone
```

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.

```
Switch(config)# mls qos map policed-dscp 24 26 46 to 0
Switch(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AutoQoS-VoIP-Control-Trust
Switch(config-cmap)# match ip dscp cs3 af31
Switch(config)# policy-map AutoQoS-Police-SoftPhone
Switch(config-pmap)# class AutoQoS-VoIP-RTP-Trust
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AutoQoS-VoIP-Control-Trust
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.

```
Switch(config-if)# service-policy input AutoQoS-Police-SoftPhone
```

If you entered the **auto qos voip cisco-phone** command, the switch automatically creates class maps and policy maps.

```
Switch(config-if)# mls qos trust device cisco-phone
```

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.

```
Switch(config)# mls qos map policed-dscp 24 26 46 to 0
Switch(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AutoQoS-VoIP-Control-Trust
Switch(config-cmap)# match ip dscp cs3 af31
Switch(config)# policy-map AutoQoS-Police-CiscoPhone
Switch(config-pmap)# class AutoQoS-VoIP-RTP-Trust
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AutoQoS-VoIP-Control-Trust
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.

```
Switch(config-if)# service-policy input AutoQoS-Police-SoftPhone
```

Examples: Auto-QoS Generated Configuration For Enhanced Video, Trust, and Classify Devices

If you entered the following enhanced auto-QoS commands, the switch configures a CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value):

- **auto qos video cts**
- **auto qos video ip-camera**
- **auto qos video media-player**
- **auto qos trust**
- **auto qos trust cos**
- **auto qos trust dscp**

The following command is initiated after entering one of the above auto-QoS commands:

```
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
```



Note No class maps and policy maps are configured.

If you entered the **auto qos classify** command, the switch automatically creates class maps and policy maps (as shown below).

```
Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
```

```

Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_MULTIENTHANCED_CONF_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENTHANCED-CONF
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Switch(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Switch(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Switch(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Switch(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICY
Switch(config-pmap)# class AUTOQOS_MULTIENTHANCED_CONF_CLASS
Switch(config-pmap-c)# set dscp af41
Switch(config-pmap-c)# class AUTOQOS_BULK_DATA_CLASS
Switch(config-pmap-c)# set dscp af11
Switch(config-pmap-c)# class AUTOQOS_TRANSACTION_CLASS
Switch(config-pmap-c)# set dscp af21
Switch(config-pmap-c)# class AUTOQOS_SCAVANGER_CLASS
Switch(config-pmap-c)# set dscp cs1
Switch(config-pmap-c)# class AUTOQOS_SIGNALING_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# class AUTOQOS_DEFAULT_CLASS
Switch(config-pmap-c)# set dscp default
;
Switch(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICY

```

If you entered the **auto qos classify police** command, the switch automatically creates class maps and policy maps (as shown below).

```

Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_MULTIENTHANCED_CONF_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENTHANCED-CONF
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Switch(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Switch(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Switch(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Switch(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICE
Switch(config-pmap)# class AUTOQOS_MULTIENTHANCED_CONF_CLASS
Switch(config-pmap-c)# set dscp af41
Switch(config-pmap-c)# police 5000000 8000 exceed-action drop
Switch(config-pmap-c)# class AUTOQOS_BULK_DATA_CLASS
Switch(config-pmap-c)# set dscp af11
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap-c)# class AUTOQOS_TRANSACTION_CLASS
Switch(config-pmap-c)# set dscp af21
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap-c)# class AUTOQOS_SCAVANGER_CLASS
Switch(config-pmap-c)# set dscp cs1
Switch(config-pmap-c)# police 10000000 8000 exceed-action drop
Switch(config-pmap-c)# class AUTOQOS_SIGNALING_CLASS
Switch(config-pmap-c)# set dscp cs3

```

```
Switch(config-pmap-c)# police 32000 8000 exceed-action drop
Switch(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Switch(config-pmap-c)# set dscp default
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
;
Switch(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
```

This is the enhanced configuration for the **auto qos voip cisco-phone** command:

```
Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-cmap)# match ip dscp cs3
Switch(config)# policy-map AUTOQOS-SRND4-CISCOPHONE-POLICY
Switch(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Switch(config-pmap-c)# set dscp default
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
;
Switch(config-if)# service-policy input AUTOQOS-SRND4-CISCOPHONE-POLICY
```

This is the enhanced configuration for the **auto qos voip cisco-softphone** command:

```
Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_MULTIENTHANCED_CONF_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENTHANCED-CONF
Switch(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Switch(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-cmap)# match ip dscp cs3
Switch(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Switch(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Switch(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER

Switch(config)# policy-map AUTOQOS-SRND4-SOFTPHONE-POLICY
Switch(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_MULTIENTHANCED_CONF_CLASS
Switch(config-pmap-c)# set dscp af41
```



```
Switch(config-pmap-c) # police 5000000 8000 exceed-action drop
Switch(config-pmap) # class AUTOQOS_BULK_DATA_CLASS
Switch(config-pmap-c) # set dscp af11
Switch(config-pmap-c) # police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap) # class AUTOQOS_TRANSACTION_CLASS
Switch(config-pmap-c) # set dscp af21
Switch(config-pmap-c) # police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap) # class AUTOQOS_SCAVANGER_CLASS
Switch(config-pmap-c) # set dscp cs1
Switch(config-pmap-c) # police 10000000 8000 exceed-action drop
Switch(config-pmap) # class AUTOQOS_SIGNALING_CLASS
Switch(config-pmap-c) # set dscp cs3
Switch(config-pmap-c) # police 32000 8000 exceed-action drop
Switch(config-pmap) # class AUTOQOS_DEFAULT_CLASS
Switch(config-pmap-c) # set dscp default
;
Switch(config-if) # service-policy input AUTOQOS-SRND4-SOFTPHONE-POLICY
```

Where to Go Next for Auto-QoS

Review the QoS documentation if you require any specific QoS changes to your auto-QoS configuration.



PART **VIII**

Routing

- [Configuring IP Unicast Routing, on page 941](#)
- [Configuring Policy-Based Routing \(PBR\), on page 1043](#)
- [Configuring EIGRP Stub Routing, on page 1049](#)



CHAPTER 47

Configuring IP Unicast Routing

- Finding Feature Information, on page 941
- Information About Configuring IP Unicast Routing, on page 941
- Information About IP Routing, on page 942
- How to Configure IP Routing, on page 942
- How to Configure IP Addressing, on page 943
- Monitoring and Maintaining IP Addressing, on page 965
- How to Configure IP Unicast Routing, on page 966
- Information About RIP, on page 967
- How to Configure RIP, on page 968
- Information About OSPF, on page 975
- Monitoring OSPF, on page 988
- Information About EIGRP, on page 989
- How to Configure EIGRP, on page 991
- Monitoring and Maintaining EIGRP, on page 999
- Information About Multi-VRF CE, on page 999
- How to Configure Multi-VRF CE, on page 1002
- Configuring Unicast Reverse Path Forwarding, on page 1019
- Protocol-Independent Features, on page 1020
- Monitoring and Maintaining the IP Network, on page 1041

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Information About Configuring IP Unicast Routing

This module describes how to configure IP Version 4 (IPv4) unicast routing on the switch.

Basic routing functions like static routing are available with . IP Base feature set and the IP Services feature set on Catalyst 3560-CX switches. Catalyst 2960-CX switches support only static routing.



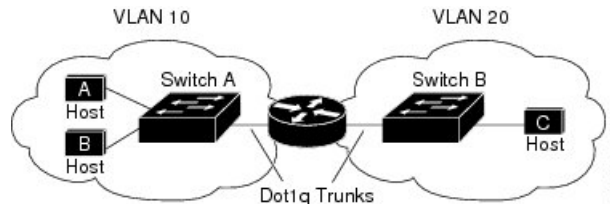
Note In addition to IPv4 traffic, you can also enable IP Version 6 (IPv6) unicast routing and configure interfaces to forward IPv6 traffic .

Information About IP Routing

In some network environments, VLANs are associated with individual networks or subnetworks. In an IP network, each subnetwork is mapped to an individual VLAN. Configuring VLANs helps control the size of the broadcast domain and keeps local traffic local. However, network devices in different VLANs cannot communicate with one another without a Layer 3 device (router) to route traffic between the VLAN, referred to as inter-VLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

Figure 83: Routing Topology Example

This figure shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router



has an interface in each VLAN.

When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet addressed to that host. Switch A forwards the packet directly to Host B, without sending it to the router.

When Host A sends a packet to Host C in VLAN 20, Switch A forwards the packet to the router, which receives the traffic on the VLAN 10 interface. The router checks the routing table, finds the correct outgoing interface, and forwards the packet on the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

Types of Routing

Routers and Layer 3 switches can route packets in these ways:

- By using default routing
- By using preprogrammed static routes for the traffic
- By dynamically calculating routes by using a routing protocol

The switch supports static routes and default routes. It does not support routing protocols.

How to Configure IP Routing

By default, IP routing is disabled on the Device, and you must enable it before routing can take place.

In the following procedures, the specified interface must be one of these Layer 3 interfaces:

- A routed port: a physical port configured as a Layer 3 port by using the **no switchport** interface configuration command.
- A switch virtual interface (SVI): a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
- An EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group.



Note The switch does not support tunnel interfaces for unicast routed traffic.

All Layer 3 interfaces on which routing will occur must have IP addresses assigned to them.



Note A Layer 3 switch can have an IP address assigned to each routed port and SVI.

Configuring routing consists of several main procedures:

- To support VLAN interfaces, create and configure VLANs on the Device or switch stack, and assign VLAN membership to Layer 2 interfaces. For more information, see the "Configuring VLANs" chapter.
- Configure Layer 3 interfaces.
- Enable IP routing on the switch.
- Assign IP addresses to the Layer 3 interfaces.
- Enable selected routing protocols on the switch.
- Configure routing protocol parameters (optional).

How to Configure IP Addressing

A required task for configuring IP routing is to assign IP addresses to Layer 3 network interfaces to enable the interfaces and allow communication with the hosts on those interfaces that use IP. The following sections describe how to configure various IP addressing features. Assigning IP addresses to the interface is required; the other procedures are optional.

- Default Addressing Configuration
- Assigning IP Addresses to Network Interfaces
- Configuring Address Resolution Methods
- Routing Assistance When IP Routing is Disabled
- Configuring Broadcast Packet Handling
- Monitoring and Maintaining IP Addressing

Default IP Addressing Configuration

Table 101: Default Addressing Configuration

Feature	Default Setting
IP address	None defined.
ARP	No permanent entries in the Address Resolution Protocol (ARP) cache. Encapsulation: Standard Ethernet-style ARP. Timeout: 14400 seconds (4 hours).
IP broadcast address	255.255.255.255 (all ones).
IP classless routing	Enabled.
IP default gateway	Disabled.
IP directed broadcast	Disabled (all IP directed broadcasts are dropped).
IP domain	Domain list: No domain names defined. Domain lookup: Enabled. Domain name: Enabled.
IP forward-protocol	If a helper address is defined or User Datagram Protocol (UDP) flooding is configured, UDP flood protection is enabled on default ports. Any-local-broadcast: Disabled. Spanning Tree Protocol (STP): Disabled. Turbo-flood: Disabled.
IP helper address	Disabled.
IP host	Disabled.
IRDP	Disabled. Defaults when enabled: <ul style="list-style-type: none"> • Broadcast IRDP advertisements. • Maximum interval between advertisements: 600 seconds. • Minimum interval between advertisements: 0.75 times max interval • Preference: 0.
IP proxy ARP	Enabled.
IP routing	Disabled.
IP subnet-zero	Disabled.

Assigning IP Addresses to Network Interfaces

An IP address identifies a location to which IP packets can be sent. Some IP addresses are reserved for special uses and cannot be used for host, subnet, or network addresses. RFC 1166, “Internet Numbers,” contains the official description of IP addresses.

An interface can have one primary IP address. A mask identifies the bits that denote the network number in an IP address. When you use the mask to subnet a network, the mask is referred to as a subnet mask. To receive an assigned network number, contact your Internet service provider.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	no switchport Example: Switch(config-if)# no switchport	Removes the interface from Layer 2 configuration mode (if it is a physical interface).
Step 5	ip address <i>ip-address subnet-mask</i> Example: Switch(config-if)# ip address 10.1.5.1 255.255.255.0	Configures the IP address and IP subnet mask.
Step 6	no shutdown Example: Switch(config-if)# no shutdown	Enables the physical interface.
Step 7	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 8	show ip route Example: Switch# show ip route	Verifies your entries.
Step 9	show ip interface [<i>interface-id</i>] Example: Switch# show ip interface gigabitethernet 1/0/1	Verifies your entries.
Step 10	show running-config Example: Switch# show running-config	Verifies your entries.
Step 11	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Using Subnet Zero

Subnetting with a subnet address of zero is strongly discouraged because of the problems that can arise if a network and a subnet have the same addresses. For example, if network 131.108.0.0 is subnetted as 255.255.255.0, subnet zero would be written as 131.108.0.0, which is the same as the network address.

You can use the all ones subnet (131.108.255.0) and even though it is discouraged, you can enable the use of subnet zero if you need the entire subnet space for your IP address.

Use the **no ip subnet-zero** global configuration command to restore the default and disable the use of subnet zero.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

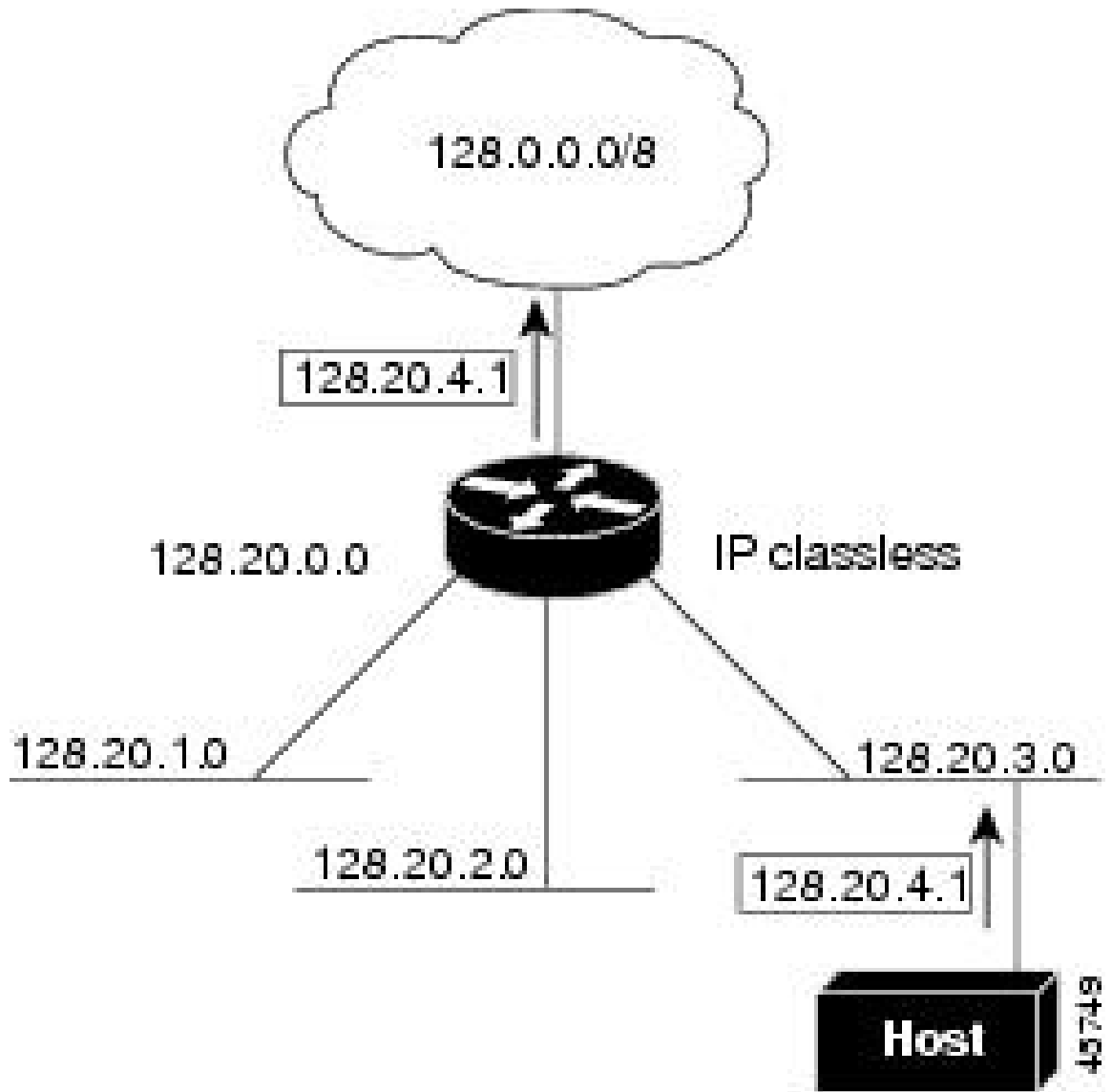
	Command or Action	Purpose
	<code>Switch# configure terminal</code>	
Step 3	ip subnet-zero Example: <code>Switch(config)# ip subnet-zero</code>	Enables the use of subnet zero for interface addresses and routing updates.
Step 4	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <code>Switch# show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Classless Routing

By default, classless routing behavior is enabled on the Device when it is configured to route. With classless routing, if a router receives packets for a subnet of a network with no default route, the router forwards the packet to the best supernet route. A supernet consists of contiguous blocks of Class C address spaces used to simulate a single, larger address space and is designed to relieve the pressure on the rapidly depleting Class B address space.

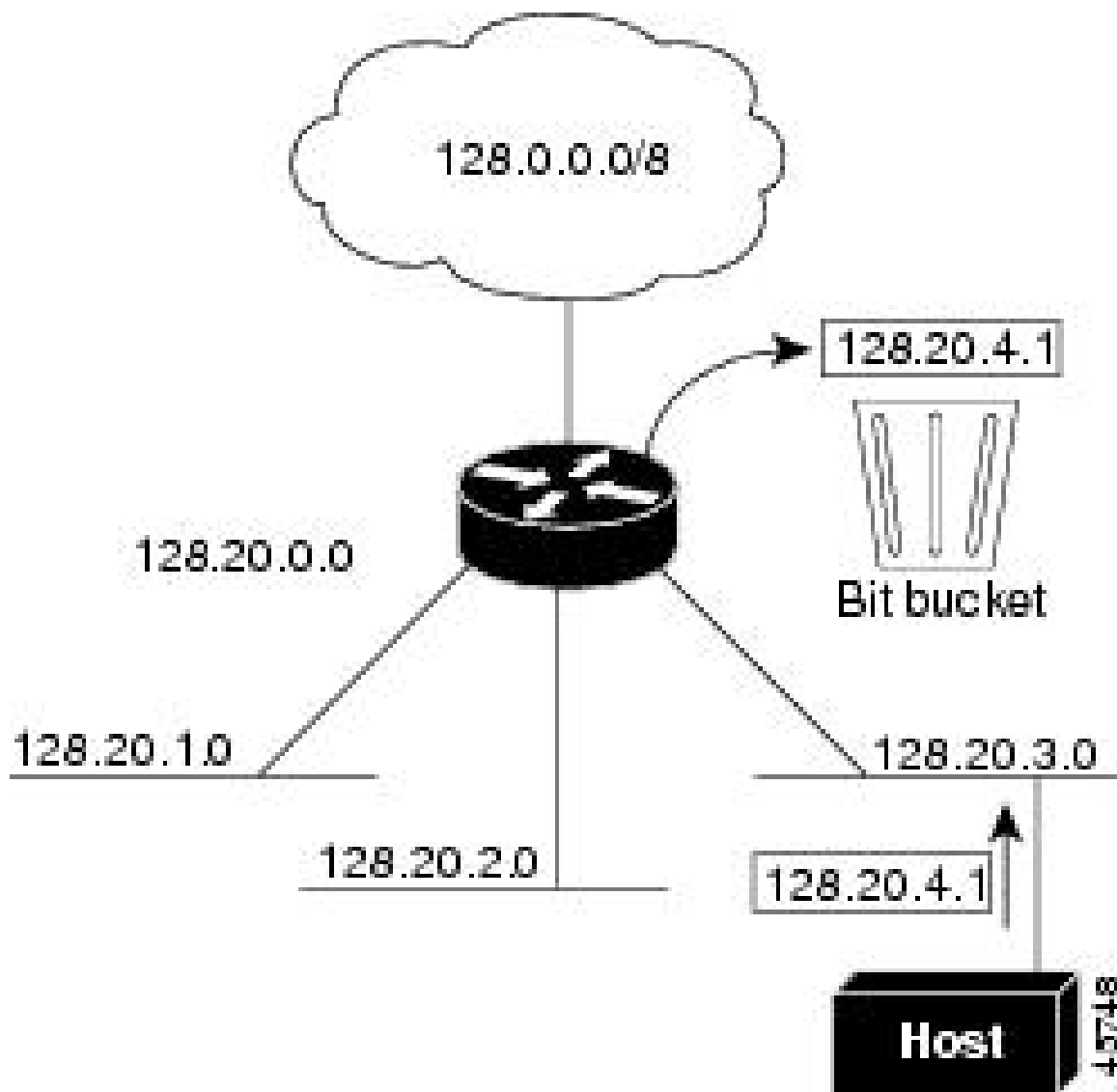
In the figure, classless routing is enabled. When the host sends a packet to 120.20.4.1, instead of discarding the packet, the router forwards it to the best supernet route. If you disable classless routing and a router receives packets destined for a subnet of a network with no network default route, the router discards the packet.

Figure 84: IP Classless Routing



In the figure, the router in network 128.20.0.0 is connected to subnets 128.20.1.0, 128.20.2.0, and 128.20.3.0. If the host sends a packet to 120.20.4.1, because there is no network default route, the router discards the packet.

Figure 85: No IP Classless Routing



To prevent the Device from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

Disabling Classless Routing

To prevent the Device from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

Procedure

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	no ip classless Example: Switch(config)# no ip classless	Disables classless routing behavior.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Address Resolution Methods

You can perform the following tasks to configure address resolution.

Address Resolution

You can control interface-specific handling of IP by using address resolution. A device using IP can have both a local address or MAC address, which uniquely defines the device on its local segment or LAN, and a network address, which identifies the network to which the device belongs.

The local address or MAC address is known as a data link address because it is contained in the data link layer (Layer 2) section of the packet header and is read by data link (Layer 2) devices. To communicate with a device on Ethernet, the software must learn the MAC address of the device. The process of learning the MAC address from an IP address is called *address resolution*. The process of learning the IP address from the MAC address is called *reverse address resolution*.

The Device can use these forms of address resolution:

- Address Resolution Protocol (ARP) is used to associate IP address with MAC addresses. Taking an IP address as input, ARP learns the associated MAC address and then stores the IP address/MAC address association in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests or replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP).
- Proxy ARP helps hosts with no routing tables learn the MAC addresses of hosts on other networks or subnets. If the Device (router) receives an ARP request for a host that is not on the same interface as the ARP request sender, and if the router has all of its routes to the host through other interfaces, it generates a proxy ARP packet giving its own local data link address. The host that sent the ARP request then sends its packets to the router, which forwards them to the intended host.

The Device also uses the Reverse Address Resolution Protocol (RARP), which functions the same as ARP does, except that the RARP packets request an IP address instead of a local MAC address. Using RARP requires a RARP server on the same network segment as the router interface. Use the **ip rarp-server address** interface configuration command to identify the server.

Defining a Static ARP Cache

ARP and other address resolution protocols provide dynamic mapping between IP addresses and MAC addresses. Because most hosts support dynamic address resolution, you usually do not need to specify static ARP cache entries. If you must define a static ARP cache entry, you can do so globally, which installs a permanent entry in the ARP cache that the Device uses to translate IP addresses into MAC addresses. Optionally, you can also specify that the Device respond to ARP requests as if it were the owner of the specified IP address. If you do not want the ARP entry to be permanent, you can specify a timeout period for the ARP entry.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	arp ip-address hardware-address type Example: Switch(config)# ip 10.1.5.1 c2f3.220a.12f4 arpa	Associates an IP address with a MAC (hardware) address in the ARP cache, and specifies encapsulation type as one of these: <ul style="list-style-type: none"> • arpa—ARP encapsulation for Ethernet interfaces • snap—Subnetwork Address Protocol encapsulation for Token Ring and FDDI interfaces • sap—HP's ARP type

	Command or Action	Purpose
Step 4	arp <i>ip-address hardware-address type</i> [alias] Example: Switch(config)# ip 10.1.5.3 d7f3.220d.12f5 arpa alias	(Optional) Specifies that the switch respond to ARP requests as if it were the owner of the specified IP address.
Step 5	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 6	arp <i>timeout seconds</i> Example: Switch(config-if)# arp 20000	(Optional) Sets the length of time an ARP cache entry will stay in the cache. The default is 14400 seconds (4 hours). The range is 0 to 2147483 seconds.
Step 7	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 8	show interfaces [<i>interface-id</i>] Example: Switch# show interfaces gigabitethernet 1/0/1	Verifies the type of ARP and the timeout value used on all interfaces or a specific interface.
Step 9	show arp Example: Switch# show arp	Views the contents of the ARP cache.
Step 10	show ip arp Example: Switch# show ip arp	Views the contents of the ARP cache.
Step 11	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting ARP Encapsulation

By default, Ethernet ARP encapsulation (represented by the **arpa** keyword) is enabled on an IP interface. You can change the encapsulation methods to SNAP if required by your network.

To disable an encapsulation type, use the **no arp arpa** or **no arp snap** interface configuration command.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	arp {arpa snap} Example: Switch(config-if)# arp arpa	Specifies the ARP encapsulation method: • arpa —Address Resolution Protocol • snap —Subnetwork Address Protocol
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show interfaces [<i>interface-id</i>] Example: Switch# show interfaces	Verifies ARP encapsulation configuration on all interfaces or the specified interface.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling Proxy ARP

By default, the Device uses proxy ARP to help hosts learn MAC addresses of hosts on other networks or subnets.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	ip proxy-arp Example: Switch(config-if)# ip proxy-arp	Enables proxy ARP on the interface.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show ip interface [<i>interface-id</i>] Example: Switch# show ip interface gigabitethernet 1/0/2	Verifies the configuration on the interface or all interfaces.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Routing Assistance When IP Routing is Disabled

These mechanisms allow the Device to learn about routes to other networks when it does not have IP routing enabled:

- Proxy ARP
- Default Gateway

- ICMP Router Discovery Protocol (IRDP)

Proxy ARP

Proxy ARP, the most common method for learning about other routes, enables an Ethernet host with no routing information to communicate with hosts on other networks or subnets. The host assumes that all hosts are on the same local Ethernet and that they can use ARP to learn their MAC addresses. If a Device receives an ARP request for a host that is not on the same network as the sender, the Device evaluates whether it has the best route to that host. If it does, it sends an ARP reply packet with its own Ethernet MAC address, and the host that sent the request sends the packet to the Device, which forwards it to the intended host. Proxy ARP treats all networks as if they are local, and performs ARP requests for every IP address.

Proxy ARP

Proxy ARP is enabled by default. To enable it after it has been disabled, see the “Enabling Proxy ARP” section. Proxy ARP works as long as other routers support it.

Default Gateway

Another method for locating routes is to define a default router or default gateway. All non-local packets are sent to this router, which either routes them appropriately or sends an IP Control Message Protocol (ICMP) redirect message back, defining which local router the host should use. The Device caches the redirect messages and forwards each packet as efficiently as possible. A limitation of this method is that there is no means of detecting when the default router has gone down or is unavailable.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip default-gateway ip-address Example: Switch(config)# ip default gateway 10.1.5.1	Sets up a default gateway (router).
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show ip redirects Example: <pre>Switch# show ip redirects</pre>	Displays the address of the default gateway router to verify the setting.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

ICMP Router Discovery Protocol

Router discovery allows the Device to dynamically learn about routes to other networks using ICMP router discovery protocol (IRDP). IRDP allows hosts to locate routers. When operating as a client, the Device generates router discovery packets. When operating as a host, the Device receives router discovery packets. The Device can also listen to Routing Information Protocol (RIP) routing updates and use this information to infer locations of routers. The Device does not actually store the routing tables sent by routing devices; it merely keeps track of which systems are sending the data. The advantage of using IRDP is that it allows each router to specify both a priority and the time after which a device is assumed to be down if no further packets are received.

Each device discovered becomes a candidate for the default router, and a new highest-priority router is selected when a higher priority router is discovered, when the current default router is declared down, or when a TCP connection is about to time out because of excessive retransmissions.

ICMP Router Discovery Protocol (IRDP)

The only required task for IRDP routing on an interface is to enable IRDP processing on that interface. When enabled, the default parameters apply.

You can optionally change any of these parameters. If you change the **maxadvertinterval** value, the **holdtime** and **minadvertinterval** values also change, so it is important to first change the **maxadvertinterval** value, before manually changing either the **holdtime** or **minadvertinterval** values.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	ip irdp Example: Switch(config-if)# ip irdp	Enables IRDP processing on the interface.
Step 5	ip irdp multicast Example: Switch(config-if)# ip irdp multicast	(Optional) Sends IRDP advertisements to the multicast address (224.0.0.1) instead of IP broadcasts. Note This command allows for compatibility with Sun Microsystems Solaris, which requires IRDP packets to be sent out as multicasts. Many implementations cannot receive these multicasts; ensure end-host ability before using this command.
Step 6	ip irdp holdtime <i>seconds</i> Example: Switch(config-if)# ip irdp holdtime 1000	(Optional) Sets the IRDP period for which advertisements are valid. The default is three times the maxadvertinterval value. It must be greater than maxadvertinterval and cannot be greater than 9000 seconds. If you change the maxadvertinterval value, this value also changes.
Step 7	ip irdp maxadvertinterval <i>seconds</i> Example: Switch(config-if)# ip irdp maxadvertinterval 650	(Optional) Sets the IRDP maximum interval between advertisements. The default is 600 seconds.
Step 8	ip irdp minadvertinterval <i>seconds</i> Example: Switch(config-if)# ip irdp minadvertinterval 500	(Optional) Sets the IRDP minimum interval between advertisements. The default is 0.75 times the maxadvertinterval . If you change the maxadvertinterval , this value changes to the new default (0.75 of maxadvertinterval).
Step 9	ip irdp preference <i>number</i> Example: Switch(config-if)# ip irdp preference 2	(Optional) Sets a device IRDP preference level. The allowed range is -231 to 231. The default is 0. A higher value increases the router preference level.
Step 10	ip irdp address <i>address [number]</i> Example: Switch(config-if)# ip irdp address 10.1.10.10	(Optional) Specifies an IRDP address and preference to proxy-advertise.

	Command or Action	Purpose
Step 11	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 12	show ip irdp Example: <pre>Switch# show ip irdp</pre>	Verifies settings by displaying IRDP values.
Step 13	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Broadcast Packet Handling

Perform the tasks in these sections to enable these schemes:

- Enabling Directed Broadcast-to-Physical Broadcast Translation
- Forwarding UDP Broadcast Packets and Protocols
- Establishing an IP Broadcast Address
- Flooding IP Broadcasts

Broadcast Packet Handling

After configuring an IP interface address, you can enable routing and configure one or more routing protocols, or you can configure the way the Device responds to network broadcasts. A broadcast is a data packet destined for all hosts on a physical network. The Device supports two kinds of broadcasting:

- A directed broadcast packet is sent to a specific network or series of networks. A directed broadcast address includes the network or subnet fields.
- A flooded broadcast packet is sent to every network.



Note You can also limit broadcast, unicast, and multicast traffic on Layer 2 interfaces by using the **storm-control** interface configuration command to set traffic suppression levels.

Routers provide some protection from broadcast storms by limiting their extent to the local cable. Bridges (including intelligent bridges), because they are Layer 2 devices, forward broadcasts to all network segments, thus propagating broadcast storms. The best solution to the broadcast storm problem is to use a single broadcast address scheme on a network. In most modern IP implementations, you can set the address to be used as the

broadcast address. Many implementations, including the one in the Device, support several addressing schemes for forwarding broadcast messages.

Enabling Directed Broadcast-to-Physical Broadcast Translation

By default, IP directed broadcasts are dropped; they are not forwarded. Dropping IP-directed broadcasts makes routers less susceptible to denial-of-service attacks.

You can enable forwarding of IP-directed broadcasts on an interface where the broadcast becomes a physical (MAC-layer) broadcast. Only those protocols configured by using the **ip forward-protocol** global configuration command are forwarded.

You can specify an access list to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts. For more information on access lists, see the “Configuring ACLs” chapter in the Security section.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/2</pre>	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip directed-broadcast [<i>access-list-number</i>] Example: <pre>Switch(config-if)# ip directed-broadcast 103</pre>	Enables directed broadcast-to-physical broadcast translation on the interface. You can include an access list to control which broadcasts are forwarded. When an access list, only IP packets permitted by the access list can be translated.
Step 5	exit Example: <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	ip forward-protocol {udp [<i>port</i>] nd sdns} Example:	Specifies which protocols and ports the router forwards when forwarding broadcast packets. <ul style="list-style-type: none"> • udp—Forward UPD datagrams.

	Command or Action	Purpose
	Switch(config)# ip forward-protocol nd	port: (Optional) Destination port that controls which UDP services are forwarded. <ul style="list-style-type: none"> • nd—Forward ND datagrams. • sdns—Forward SDNS datagrams
Step 7	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 8	show ip interface [<i>interface-id</i>] Example: Switch# show ip interface	Verifies the configuration on the interface or all interfaces
Step 9	show running-config Example: Switch# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

UDP Broadcast Packets and Protocols

User Datagram Protocol (UDP) is an IP host-to-host layer protocol, as is TCP. UDP provides a low-overhead, connectionless session between two end systems and does not provide for acknowledgment of received datagrams. Network hosts occasionally use UDP broadcasts to find address, configuration, and name information. If such a host is on a network segment that does not include a server, UDP broadcasts are normally not forwarded. You can remedy this situation by configuring an interface on a router to forward certain classes of broadcasts to a helper address. You can use more than one helper address per interface.

You can specify a UDP destination port to control which UDP services are forwarded. You can specify multiple UDP protocols. You can also specify the Network Disk (ND) protocol, which is used by older diskless Sun workstations and the network security protocol SDNS.

By default, both UDP and ND forwarding are enabled if a helper address has been defined for an interface.

Forwarding UDP Broadcast Packets and Protocols

If you do not specify any UDP ports when you configure the forwarding of UDP broadcasts, you are configuring the router to act as a BOOTP forwarding agent. BOOTP packets carry DHCP information.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	ip helper-address <i>address</i> Example: <pre>Switch(config-if)# ip helper address 10.1.10.1</pre>	Enables forwarding and specifies the destination address for forwarding UDP broadcast packets, including BOOTP.
Step 5	exit Example: <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	ip forward-protocol {udp [<i>port</i>] nd sdns} Example: <pre>Switch(config)# ip forward-protocol sdns</pre>	Specifies which protocols the router forwards when forwarding broadcast packets.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	show ip interface [<i>interface-id</i>] Example: <pre>Switch# show ip interface gigabitethernet 1/0/1</pre>	Verifies the configuration on the interface or all interfaces.
Step 9	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show running-config</code>	
Step 10	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Establishing an IP Broadcast Address

The most popular IP broadcast address (and the default) is an address consisting of all ones (255.255.255.255). However, the Device can be configured to generate any form of IP broadcast address.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip broadcast-address <i>ip-address</i> Example: Switch(config-if)# <code>ip broadcast-address 128.1.255.255</code>	Enters a broadcast address different from the default, for example 128.1.255.255.
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show ip interface [<i>interface-id</i>] Example: <pre>Switch# show ip interface</pre>	Verifies the broadcast address on the interface or all interfaces.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

IP Broadcast Flooding

You can allow IP broadcasts to be flooded throughout your internetwork in a controlled fashion by using the database created by the bridging STP. Using this feature also prevents loops. To support this capability, bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, it still can receive broadcasts. However, the interface never forwards broadcasts it receives, and the router never uses that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper-address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

To be considered for flooding, packets must meet these criteria. (Note that these are the same conditions used to consider packet forwarding using IP helper addresses.)

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast.
- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP specified by the **ip forward-protocol udp** global configuration command.
- The time-to-live (TTL) value of the packet must be at least two.

A flooded UDP datagram is given the destination address specified with the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any address. Thus, the destination address might change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

When a flooded UDP datagram is sent out an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is, therefore, subject to access lists, if they are present on the output interface.

In the Device, the majority of packets are forwarded in hardware; most packets do not go through the Device CPU. For those packets that do go to the CPU, you can speed up spanning tree-based UDP flooding by a factor of about four to five times by using turbo-flooding. This feature is supported over Ethernet interfaces configured for ARP encapsulation.

Flooding IP Broadcasts

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip forward-protocol spanning-tree Example: Switch(config)# ip forward-protocol spanning-tree	Uses the bridging spanning-tree database to flood UDP datagrams.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.
Step 7	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 8	ip forward-protocol turbo-flood Example:	Uses the spanning-tree database to speed up flooding of UDP datagrams.

	Command or Action	Purpose
	<code>Switch(config)# ip forward-protocol turbo-flood</code>	
Step 9	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.
Step 10	show running-config Example: <code>Switch# show running-config</code>	Verifies your entries.
Step 11	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining IP Addressing

When the contents of a particular cache, table, or database have become or are suspected to be invalid, you can remove all its contents by using the **clear** privileged EXEC commands. The Table lists the commands for clearing contents.

Table 102: Commands to Clear Caches, Tables, and Databases

clear arp-cache	Clears the IP ARP cache and the fast-switching cache.
clear host { <i>name</i> *}	Removes one or all entries from the hostname and the address.
clear ip route { <i>network</i> [<i>mask</i>] *}	Removes one or more routes from the IP routing table.

You can display specific statistics, such as the contents of IP routing tables, caches, and databases; the reachability of nodes; and the routing path that packets are taking through the network. The Table lists the privileged EXEC commands for displaying IP statistics.

Table 103: Commands to Display Caches, Tables, and Databases

show arp	Displays the entries in the ARP table.
show hosts	Displays the default domain name, style of lookup service, name server, and the cached list of hostnames and addresses.
show ip aliases	Displays IP addresses mapped to TCP ports (aliases).
show ip arp	Displays the IP ARP cache.

show ip interface [<i>interface-id</i>]	Displays the IP status of interfaces.
show ip irdp	Displays IRDP values.
show ip masks <i>address</i>	Displays the masks used for network addresses and the number of subnets for each mask.
show ip redirects	Displays the address of a default gateway.
show ip route [<i>address</i> [<i>mask</i>]] [<i>protocol</i>]	Displays the current state of the routing table.
show ip route summary	Displays the current state of the routing table in summary form.

How to Configure IP Unicast Routing

Enabling IP Unicast Routing

By default, the Device is in Layer 2 switching mode and IP routing is disabled. To use the Layer 3 capabilities of the Device, you must enable IP routing.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip routing Example: Switch(config)# ip routing	Enables IP routing.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Example of Enabling IP Unicast Routing

This example shows how to enable IP routing on a Switch:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# ip routing

Switch(config-router)# end
```

Information About RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) created for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The protocol is documented in RFC 1058. You can find detailed information about RIP in *IP Routing Fundamentals*, published by Cisco Press.



Note RIP is supported in the Network Essentials feature set.

Using RIP, the Device sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the non-updating router.

RIP uses hop counts to rate the value of different routes. The hop count is the number of routers that can be traversed in a route. A directly connected network has a hop count of zero; a network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudonetwork 0.0.0.0. The 0.0.0.0 network does not exist; it is treated by RIP as a network to implement the default routing feature. The Device advertises the default network if a default was learned by RIP or if the router has a gateway of last resort and RIP is configured with a default metric. RIP sends updates to the interfaces in specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

How to Configure RIP

Default RIP Configuration

Table 104: Default RIP Configuration

Feature	Default Setting
Auto summary	Enabled.
Default-information originate	Disabled.
Default metric	Built-in; automatic metric translations.
IP RIP authentication key-chain	No authentication. Authentication mode: clear text.
IP RIP triggered	Disabled
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.
Timers basic	<ul style="list-style-type: none"> • Update: 30 seconds. • Invalid: 180 seconds. • Hold-down: 180 seconds. • Flush: 240 seconds.
Validate-update-source	Enabled.
Version	Receives RIP Version 1 and 2 packets; sends Version 1 packets.

Configuring Basic RIP Parameters

To configure RIP, you enable RIP routing for a network and optionally configure other parameters. On the Device, RIP configuration commands are ignored until you configure the network number.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip routing Example: Switch(config)# ip routing	Enables IP routing. (Required only if IP routing is disabled.)
Step 4	router rip Example: Switch(config)# router rip	Enables a RIP routing process, and enter router configuration mode.
Step 5	network <i>network number</i> Example: Switch(config-router)# network 12.0.0.0	Associates a network with a RIP routing process. You can specify multiple network commands. RIP routing updates are sent and received through interfaces only on these networks. Note You must configure a network number for the RIP commands to take effect.
Step 6	neighbor <i>ip-address</i> Example: Switch(config-router)# neighbor 10.2.5.1	(Optional) Defines a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 7	offset-list [<i>access-list number name</i>] {<i>in out</i>} <i>offset</i> [<i>type number</i>] Example: Switch(config-router)# offset-list 103 in 10	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.
Step 8	timers <i>basic update invalid holddown flush</i> Example: Switch(config-router)# timers basic 45 360 400 300	(Optional) Adjusts routing protocol timers. Valid ranges for all timers are 0 to 4294967295 seconds. <ul style="list-style-type: none"> • <i>update</i>—The time between sending routing updates. The default is 30 seconds.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>invalid</i>—The timer after which a route is declared invalid. The default is 180 seconds. • <i>holddown</i>—The time before a route is removed from the routing table. The default is 180 seconds. • <i>flush</i>—The amount of time for which routing updates are postponed. The default is 240 seconds.
Step 9	version {1 2} Example: <pre>Switch(config-router)# version 2</pre>	(Optional) Configures the switch to receive and send only RIP Version 1 or RIP Version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands ip rip {send receive} version 1 2 1 2 to control what versions are used for sending and receiving on interfaces.
Step 10	no auto-summary Example: <pre>Switch(config-router)# no auto-summary</pre>	(Optional) Disables automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disable summarization (RIP Version 2 only) to advertise subnet and host routing information to classful network boundaries.
Step 11	output-delay delay Example: <pre>Switch(config-router)# output-delay 8</pre>	(Optional) Adds interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.
Step 12	end Example: <pre>Switch(config-router)# end</pre>	Returns to privileged EXEC mode.
Step 13	show ip protocols Example: <pre>Switch# show ip protocols</pre>	Verifies your entries.
Step 14	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring RIP Authentication

RIP Version 1 does not support authentication. If you are sending and receiving RIP Version 2 packets, you can enable RIP authentication on an interface. The key chain specifies the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed, not even the default.

The Device supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip rip authentication key-chain <i>name-of-chain</i> Example: <pre>Switch(config-if)# ip rip authentication key-chain trees</pre>	Enables RIP authentication.
Step 5	ip rip authentication mode {text md5} Example: <pre>Switch(config-if)# ip rip authentication mode md5</pre>	Configures the interface to use plain text authentication (the default) or MD5 digest authentication.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Summary Addresses and Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature usually optimizes communication among multiple routers, especially when links are broken.

Configuring Summary Addresses and Split Horizon



Note In general, disabling split horizon is not recommended unless you are certain that your application requires it to properly advertise routes.

If you want to configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.



Note If split horizon is enabled, neither autosummary nor interface IP summary addresses are advertised.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.

	Command or Action	Purpose
Step 4	ip address <i>ip-address subnet-mask</i> Example: <pre>Switch(config-if)# ip address 10.1.1.10 255.255.255.0</pre>	Configures the IP address and IP subnet.
Step 5	ip summary-address rip <i>ip-address ip-network mask</i> Example: <pre>Switch(config-if)# ip summary-address rip 10.1.1.30 255.255.255.0</pre>	Configures the IP address to be summarized and the IP network mask.
Step 6	no ip split-horizon Example: <pre>Switch(config-if)# no ip split-horizon</pre>	Disables split horizon on the interface.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	show ip interface <i>interface-id</i> Example: <pre>Switch# show ip interface gigabitethernet 1/0/1</pre>	Verifies your entries.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature can optimize communication among multiple routers, especially when links are broken.



Note In general, we do not recommend disabling split horizon unless you are certain that your application requires it to properly advertise routes.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip address <i>ip-address subnet-mask</i> Example: Switch(config-if)# ip address 10.1.1.10 255.255.255.0	Configures the IP address and IP subnet.
Step 5	no ip split-horizon Example: Switch(config-if)# no ip split-horizon	Disables split horizon on the interface.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show ip interface <i>interface-id</i> Example: Switch# show ip interface gigabitethernet 1/0/1	Verifies your entries.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuration Example for Summary Addresses and Split Horizon

In this example, the major net is 10.0.0.0. The summary address 10.2.0.0 overrides the autosummary address of 10.0.0.0 so that 10.2.0.0 is advertised out interface Gigabit Ethernet port 2, and 10.0.0.0 is not advertised. In the example, if the interface is still in Layer 2 mode (the default), you must enter a **no switchport** interface configuration command before entering the **ip address** interface configuration command.



Note If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip** router configuration command) are advertised.

```
Switch(config)# router rip
Switch(config-router)# interface gigabitethernet1/0/2
Switch(config-if)# ip address 10.1.5.1 255.255.255.0
Switch(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
Switch(config-if)# no ip split-horizon
Switch(config-if)# exit
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# neighbor 2.2.2.2 peer-group mygroup
Switch(config-router)# end
```

Information About OSPF

OSPF is an Interior Gateway Protocol (IGP) designed expressly for IP networks, supporting IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets. The Cisco implementation supports RFC 1253, OSPF management information base (MIB).

The Cisco implementation conforms to the OSPF Version 2 specifications with these key features:

- Definition of stub areas is supported.
- Routes learned through any IP routing protocol can be redistributed into another IP routing protocol. At the intradomain level, this means that OSPF can import routes learned through EIGRP and RIP. OSPF routes can also be exported into RIP.
- Plain text and MD5 authentication among neighboring routers within an area is supported.
- Configurable routing interface parameters include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Virtual links are supported.
- Not-so-stubby-areas (NSSAs) per RFC 1587 are supported.

OSPF typically requires coordination among many internal routers, area border routers (ABRs) connected to multiple areas, and autonomous system boundary routers (ASBRs). The minimum configuration would use all default parameter values, no authentication, and interfaces assigned to areas. If you customize your environment, you must ensure coordinated configuration of all routers.

How to Configure OSPF

Default OSPF Configuration

Table 105: Default OSPF Configuration

Feature	Default Setting
Interface parameters	Cost: No default cost predefined Retransmit interval: 5 seconds. Transmit delay: 1 second. Priority: 1. Hello interval: 10 seconds. Dead interval: 4 times the hello interval. No authentication. No password specified. MD5 authentication disabled.
Area	Authentication type: 0 (no authentication). Default cost: 1. Range: Disabled. Stub: No stub area defined. NSSA: No NSSA area defined.
Auto cost	100 Mb/s.
Default-information originate	Disabled. When enabled, the default metric setting is 10, and the external route type is Type 2.
Default metric	Built-in, automatic metric translation, as appropriate for each routing protocol.
Distance OSPF	dist1 (all routes within an area): 110. dist2 (all routes from one area to another): 110. dist3 (routes from other routing domains): 110.
OSPF database filter	Disabled. All outgoing link-state advertisements (LSAs) are flooded to the interface.
IP OSPF name lookup	Disabled.
Log adjacency changes	Enabled.
Neighbor	None specified.
Neighbor database filter	Disabled. All outgoing LSAs are flooded to the neighbor.
Network area	Disabled.

Feature	Default Setting
Nonstop Forwarding (NSF) awareness	Enabled. Allows Layer 3 Device to continue forwarding packets from a neighbor NSF-capable router during hardware or software changes.
NSF capability	Disabled. Note The Device stack supports OSPF NSF-capable routing for IPv4.
Router ID	No OSPF routing process defined.
Summary address	Disabled.
Timers LSA group pacing	240 seconds.
Timers shortest path first (spf)	spf delay: 5 seconds.; spf-holdtime: 10 seconds.
Virtual link	No area ID or router ID defined. Hello interval: 10 seconds. Retransmit interval: 5 seconds. Transmit delay: 1 second. Dead interval: 40 seconds. Authentication key: no key predefined. Message-digest key (MD5): no key predefined.

OSPF for Routed Access

With Cisco IOS Release 12.2(55)SE, the IP Base image supports OSPF for routed access. The IP services image is required if you need multiple OSPFv2 and OSPFv3 instances without route restrictions. Additionally, the IP services image is required to enable the multi-VRF-CE feature.

OSPF for Routed Access is specifically designed so that you can extend Layer 3 routing capabilities to the wiring closet.



Note OSPF for Routed Access supports only one OSPFv2 and one OSPFv3 instance with a combined total of 1000 dynamically learned routes. The IP Base image provides OSPF for routed access.

However, these restrictions are not enforced in this release.

With the typical topology (hub and spoke) in a campus environment, where the wiring closets (spokes) are connected to the distribution switch (hub) that forwards all nonlocal traffic to the distribution layer, the wiring closet Device need not hold a complete routing table. A best practice design, where the distribution Device sends a default route to the wiring closet Device to reach interarea and external routes (OSPF stub or totally stub area configuration) should be used when OSPF for Routed Access is used in the wiring closet.

For more details, see the “High Availability Campus Network Design—Routed Access Layer using EIGRP or OSPF” document.

OSPF Nonstop Forwarding

The Device or switch stack supports two levels of nonstop forwarding (NSF):

- [OSPF NSF Awareness, on page 978](#)
- [OSPF NSF Capability, on page 978](#)

OSPF NSF Awareness

When the neighboring router is NSF-capable, the Layer 3 Device continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router crashing and the backup RP taking over, or while the primary RP is manually reloaded for a non-disruptive software upgrade.

This feature cannot be disabled.

OSPF NSF Capability

supports the OSPFv2 NSF IETF format in addition to the OSPFv2 NSF Cisco format that is supported in earlier releases. For information about this feature, see : *NSF—OSPF (RFC 3623 OSPF Graceful Restart)*.

The also supports OSPF NSF-capable routing for IPv4 for better convergence and lower traffic loss following a stack's active switch change.



Note OSPF NSF requires that all neighbor networking devices be NSF-aware. If an NSF-capable router discovers non-NSF aware neighbors on a network segment, it disables NSF capabilities for that segment. Other network segments where all devices are NSF-aware or NSF-capable continue to provide NSF capabilities.

Use the **nsf** OSPF routing configuration command to enable OSPF NSF routing. Use the **show ip ospf** privileged EXEC command to verify that it is enabled.

For more information, see *Cisco Nonstop Forwarding*:

http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-nonstp_fwdg.html

Configuring Basic OSPF Parameters

To enable OSPF, create an OSPF routing process, specify the range of IP addresses to associate with the routing process, and assign area IDs to be associated with that range.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router ospf process-id Example: Switch(config)# router ospf 15	Enables OSPF routing, and enter router configuration mode. The process ID is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value.

	Command or Action	Purpose
		Note OSPF for Routed Access supports only one OSPFv2 and one OSPFv3 instance with a maximum number of 1000 dynamically learned routes.
Step 3	network <i>address wildcard-mask</i> area <i>area-id</i> Example: Switch(config-router)# network 10.1.1.1 255.240.0.0 area 20	Define an interface on which OSPF runs and the area ID for that interface. You can use the wildcard-mask to use a single command to define one or more multiple interfaces to be associated with a specific OSPF area. The area ID can be a decimal value or an IP address.
Step 4	end Example: Switch(config-router)# end	Returns to privileged EXEC mode.
Step 5	show ip protocols Example: Switch# show ip protocols	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Example: Configuring Basic OSPF Parameters

This example shows how to configure an OSPF routing process and assign it a process number of 109:

```
Switch(config)# router ospf 109
Switch(config-router)# network 131.108.0.0 255.255.255.0 area 24
```

Configuring OSPF Interfaces

You can use the **ip ospf** interface configuration commands to modify interface-specific OSPF parameters. You are not required to modify any of these parameters, but some interface parameters (hello interval, dead interval, and authentication key) must be consistent across all routers in an attached network. If you modify these parameters, be sure all routers in the network have compatible values.



Note The **ip ospf** interface configuration commands are all optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	ip ospf cost cost Example: Switch(config-if)# ip ospf cost 8	(Optional) Explicitly specifies the cost of sending a packet on the interface.
Step 4	ip ospf retransmit-interval seconds Example: Switch(config-if)# ip ospf transmit-interval 10	(Optional) Specifies the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
Step 5	ip ospf transmit-delay seconds Example: Switch(config-if)# ip ospf transmit-delay 2	(Optional) Sets the estimated number of seconds to wait before sending a link state update packet. The range is 1 to 65535 seconds. The default is 1 second.
Step 6	ip ospf priority number Example: Switch(config-if)# ip ospf priority 5	(Optional) Sets priority to help find the OSPF designated router for a network. The range is from 0 to 255. The default is 1.
Step 7	ip ospf hello-interval seconds Example: Switch(config-if)# ip ospf hello-interval 12	(Optional) Sets the number of seconds between hello packets sent on an OSPF interface. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 10 seconds.
Step 8	ip ospf dead-interval seconds Example: Switch(config-if)# ip ospf dead-interval 8	(Optional) Sets the number of seconds after the last device hello packet was seen before its neighbors declare the OSPF router to be down. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 4 times the hello interval.
Step 9	ip ospf authentication-key key Example:	(Optional) Assign a password to be used by neighboring OSPF routers. The password can be any string of keyboard-entered characters up to 8 bytes in length. All

	Command or Action	Purpose
	Switch(config-if)# ip ospf authentication-key password	neighboring routers on the same network must have the same password to exchange OSPF information.
Step 10	ip ospf message-digest-key <i>keyid</i> md5 <i>key</i> Example: Switch(config-if)# ip ospf message digest-key 16 md5 your1pass	(Optional) Enables MDS authentication. <ul style="list-style-type: none"> • <i>keyid</i>—An identifier from 1 to 255. • <i>key</i>—An alphanumeric password of up to 16 bytes.
Step 11	ip ospf database-filter all out Example: Switch(config-if)# ip ospf database-filter all out	(Optional) Block flooding of OSPF LSA packets to the interface. By default, OSPF floods new LSAs over all interfaces in the same area, except the interface on which the LSA arrives.
Step 12	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 13	show ip ospf interface [<i>interface-name</i>] Example: Switch# show ip ospf interface	Displays OSPF-related interface information.
Step 14	show ip ospf neighbor detail Example: Switch# show ip ospf neighbor detail	Displays NSF awareness status of neighbor switch. The output matches one of these examples: <ul style="list-style-type: none"> • <i>Options is 0x52</i> <i>LLS Options is 0x1 (LR)</i> When both of these lines appear, the neighbor switch is NSF aware. • <i>Options is 0x42</i>—This means the neighbor switch is not NSF aware.
Step 15	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

OSPF Area Parameters

You can optionally configure several OSPF area parameters. These parameters include authentication for password-based protection against unauthorized access to an area, stub areas, and not-so-stubby-areas (NSSAs). Stub areas are areas into which information on external routes is not sent. Instead, the area border router (ABR)

generates a default external route into the stub area for destinations outside the autonomous system (AS). An NSSA does not flood all LSAs from the core into the area, but can import AS external routes within the area by redistribution.

Route summarization is the consolidation of advertised addresses into a single summary route to be advertised by other areas. If network numbers are contiguous, you can use the **area range** router configuration command to configure the ABR to advertise a summary route that covers all networks in the range.

Configuring OSPF Area Parameters

Before you begin



Note The OSPF **area** router configuration commands are all optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router ospf process-id Example: Switch(config)# router ospf 109	Enables OSPF routing, and enter router configuration mode.
Step 3	area area-id authentication Example: Switch(config-router)# area 1 authentication	(Optional) Allow password-based protection against unauthorized access to the identified area. The identifier can be either a decimal value or an IP address.
Step 4	area area-id authentication message-digest Example: Switch(config-router)# area 1 authentication message-digest	(Optional) Enables MD5 authentication on the area.
Step 5	area area-id stub [no-summary] Example: Switch(config-router)# area 1 stub	(Optional) Define an area as a stub area. The no-summary keyword prevents an ABR from sending summary link advertisements into the stub area.
Step 6	area area-id nssa [no-redistribution] [default-information-originate] [no-summary] Example:	(Optional) Defines an area as a not-so-stubby-area. Every router within the same area must agree that the area is NSSA. Select one of these keywords:

	Command or Action	Purpose
	<pre>Switch(config-router)# area 1 nssa default-information-originate</pre>	<ul style="list-style-type: none"> • no-redistribution—Select when the router is an NSSA ABR and you want the redistribute command to import routes into normal areas, but not into the NSSA. • default-information-originate—Select on an ABR to allow importing type 7 LSAs into the NSSA. • no-redistribution—Select to not send summary LSAs into the NSSA.
Step 7	<p>area <i>area-id</i> range <i>address mask</i></p> <p>Example:</p> <pre>Switch(config-router)# area 1 range 255.240.0.0</pre>	(Optional) Specifies an address range for which a single route is advertised. Use this command only with area border routers.
Step 8	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	<p>show ip ospf [<i>process-id</i>]</p> <p>Example:</p> <pre>Switch# show ip ospf</pre>	Displays information about the OSPF routing process in general or for a specific process ID to verify configuration.
Step 10	<p>show ip ospf [<i>process-id</i> [<i>area-id</i>]] database</p> <p>Example:</p> <pre>Switch# show ip ospf database</pre>	Displays lists of information related to the OSPF database for a specific router.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Other OSPF Parameters

You can optionally configure other OSPF parameters in router configuration mode.

- **Route summarization:** When redistributing routes from other protocols. Each route is advertised individually in an external LSA. To help decrease the size of the OSPF link state database, you can use the **summary-address** router configuration command to advertise a single router for all the redistributed routes included in a specified network address and mask.
- **Virtual links:** In OSPF, all areas must be connected to a backbone area. You can establish a virtual link in case of a backbone-continuity break by configuring two Area Border Routers as endpoints of a virtual link. Configuration information includes the identity of the other virtual endpoint (the other ABR) and

the nonbackbone link that the two routers have in common (the transit area). Virtual links cannot be configured through a stub area.

- **Default route:** When you specifically configure redistribution of routes into an OSPF routing domain, the route automatically becomes an autonomous system boundary router (ASBR). You can force the ASBR to generate a default route into the OSPF routing domain.
- **Domain Name Server (DNS) names for use in all OSPF `show` privileged EXEC command displays** makes it easier to identify a router than displaying it by router ID or neighbor ID.
- **Default Metrics:** OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The metric is calculated as *ref-bw* divided by bandwidth, where *ref* is 10 by default, and bandwidth (*bw*) is specified by the **bandwidth** interface configuration command. For multiple links with high bandwidth, you can specify a larger number to differentiate the cost on those links.
- **Administrative distance** is a rating of the trustworthiness of a routing information source, an integer between 0 and 255, with a higher value meaning a lower trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored. OSPF uses three different administrative distances: routes within an area (interarea), routes to another area (interarea), and routes from another routing domain learned through redistribution (external). You can change any of the distance values.
- **Passive interfaces:** Because interfaces between two devices on an Ethernet represent only one network segment, to prevent OSPF from sending hello packets for the sending interface, you must configure the sending device to be a passive interface. Both devices can identify each other through the hello packet for the receiving interface.
- **Route calculation timers:** You can configure the delay time between when OSPF receives a topology change and when it starts the shortest path first (SPF) calculation and the hold time between two SPF calculations.
- **Log neighbor changes:** You can configure the router to send a syslog message when an OSPF neighbor state changes, providing a high-level view of changes in the router.

Configuring Other OSPF Parameters

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	router ospf <i>process-id</i> Example: Switch(config)# <code>router ospf 10</code>	Enables OSPF routing, and enter router configuration mode.

	Command or Action	Purpose
Step 3	summary-address <i>address mask</i> Example: <pre>Switch(config)# summary-address 10.1.1.1 255.255.255.0</pre>	(Optional) Specifies an address and IP subnet mask for redistributed routes so that only one summary route is advertised.
Step 4	area <i>area-id</i> virtual-link <i>router-id</i> [hello-interval <i>seconds</i>] [retransmit-interval <i>seconds</i>] [trans [[authentication-key <i>key</i>] message-digest-key <i>keyid md5 key</i>]] Example: <pre>Switch(config)# area 2 virtual-link 192.168.255.1 hello-interval 5</pre>	(Optional) Establishes a virtual link and set its parameters.
Step 5	default-information originate [always] [metric <i>metric-value</i>] [metric-type <i>type-value</i>] [route-map <i>map-name</i>] Example: <pre>Switch(config)# default-information originate metric 100 metric-type 1</pre>	(Optional) Forces the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
Step 6	ip ospf name-lookup Example: <pre>Switch(config)# ip ospf name-lookup</pre>	(Optional) Configures DNS name lookup. The default is disabled.
Step 7	ip auto-cost reference-bandwidth <i>ref-bw</i> Example: <pre>Switch(config)# ip auto-cost reference-bandwidth 5</pre>	(Optional) Specifies an address range for which a single route will be advertised. Use this command only with area border routers.
Step 8	distance ospf {[inter-area <i>dist1</i>] [inter-area <i>dist2</i>] [external <i>dist3</i>]} Example: <pre>Switch(config)# distance ospf inter-area 150</pre>	(Optional) Changes the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.
Step 9	passive-interface <i>type number</i> Example: <pre>Switch(config)# passive-interface gigabitethernet 1/0/6</pre>	(Optional) Suppresses the sending of hello packets through the specified interface.
Step 10	timers throttle spf <i>spf-delay</i> <i>spf-holdtime</i> <i>spf-wait</i> Example:	(Optional) Configures route calculation timers.

	Command or Action	Purpose
	Switch(config)# <code>timers throttle spf 200 100 100</code>	<ul style="list-style-type: none"> • <i>spf-delay</i>—Delay between receiving a change to SPF calculation. The range is from 1 to 600000 in milliseconds. • <i>spf-holdtime</i>—Delay between first and second SPF calculation. The range is from 1 to 600000 in milliseconds. • <i>spf-wait</i>—Maximum wait time in milliseconds for SPF calculations. The range is from 1 to 600000 in milliseconds.
Step 11	ospf log-adj-changes Example: Switch(config)# <code>ospf log-adj-changes</code>	(Optional) Sends syslog message when a neighbor state changes.
Step 12	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 13	show ip ospf [process-id [area-id]] database Example: Switch# <code>show ip ospf database</code>	Displays lists of information related to the OSPF database for a specific router.
Step 14	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

LSA Group Pacing

The OSPF LSA group pacing feature allows the router to group OSPF LSAs and pace the refreshing, check-summing, and aging functions for more efficient router use. This feature is enabled by default with a 4-minute default pacing interval, and you will not usually need to modify this parameter. The optimum group pacing interval is inversely proportional to the number of LSAs the router is refreshing, check-summing, and aging. For example, if you have approximately 10,000 LSAs in the database, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

Changing LSA Group Pacing

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	router ospf <i>process-id</i> Example: Switch(config)# <code>router ospf 25</code>	Enables OSPF routing, and enter router configuration mode.
Step 3	timers lsa-group-pacing <i>seconds</i> Example: Switch(config-router)# <code>timers lsa-group-pacing 15</code>	Changes the group pacing of LSAs.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Loopback Interfaces

OSPF uses the highest IP address configured on the interfaces as its router ID. If this interface is down or removed, the OSPF process must recalculate a new router ID and resend all its routing information out its interfaces. If a loopback interface is configured with an IP address, OSPF uses this IP address as its router ID, even if other interfaces have higher IP addresses. Because loopback interfaces never fail, this provides greater stability. OSPF automatically prefers a loopback interface over other interfaces, and it chooses the highest IP address among all loopback interfaces.

Configuring a Loopback Interface

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface loopback 0 Example: Switch(config)# <code>interface loopback 0</code>	Creates a loopback interface, and enter interface configuration mode.
Step 3	ip address address mask Example: Switch(config-if)# <code>ip address 10.1.1.5 255.255.240.0</code>	Assign an IP address to this interface.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show ip interface Example: Switch# <code>show ip interface</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Table 106: Show IP OSPF Statistics Commands

<code>show ip ospf [process-id]</code>	Displays general information about OSPF processes.
----------------------------------------	----------------------------------------------------

<pre>show ip ospf [process-id] database [router] [link-state-id] show ip ospf [process-id] database [router] [self-originate] show ip ospf [process-id] database [router] [adv-router [ip-address]] show ip ospf [process-id] database [network] [link-state-id] show ip ospf [process-id] database [summary] [link-state-id] show ip ospf [process-id] database [asbr-summary] [link-state-id] show ip ospf [process-id] database [external] [link-state-id] show ip ospf [process-id area-id] database [database-summary]</pre>	Displays lists of information.
<pre>show ip ospf border-routes</pre>	Displays the internal entries.
<pre>show ip ospf interface [interface-name]</pre>	Displays OSPF-related information.
<pre>show ip ospf neighbor [interface-name] [neighbor-id] detail</pre>	Displays OSPF neighbor information.
<pre>show ip ospf virtual-links</pre>	Displays OSPF-related information.

Information About EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the IGRP. EIGRP uses the same distance vector algorithm and distance information as IGRP; however, the convergence properties and the operating efficiency of EIGRP are significantly improved.

The convergence technology employs an algorithm referred to as the Diffusing Update Algorithm (DUAL), which guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

IP EIGRP provides increased network width. With RIP, the largest possible width of your network is 15 hops. Because the EIGRP metric is large enough to support thousands of hops, the only barrier to expanding the network is the transport-layer hop counter. EIGRP increments the transport control field only when an IP packet has traversed 15 routers and the next hop to the destination was learned through EIGRP. When a RIP route is used as the next hop to the destination, the transport control field is incremented as usual.

EIGRP Features

EIGRP offers these features:

- Fast convergence.
- Incremental updates when the state of a destination changes, instead of sending the entire contents of the routing table, minimizing the bandwidth required for EIGRP packets.
- Less CPU usage because full update packets need not be processed each time they are received.
- Protocol-independent neighbor discovery mechanism to learn about neighboring routers.

- Variable-length subnet masks (VLSMs).
- Arbitrary route summarization.
- EIGRP scales to large networks.

EIGRP Components

EIGRP has these four basic components:

- Neighbor discovery and recovery is the process that routers use to dynamically learn of other routers on their directly attached networks. Routers must also discover when their neighbors become unreachable or inoperative. Neighbor discovery and recovery is achieved with low overhead by periodically sending small hello packets. As long as hello packets are received, the Cisco IOS software can learn that a neighbor is alive and functioning. When this status is determined, the neighboring routers can exchange routing information.
- The reliable transport protocol is responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors. It supports intermixed transmission of multicast and unicast packets. Some EIGRP packets must be sent reliably, and others need not be. For efficiency, reliability is provided only when necessary. For example, on a multiaccess network that has multicast capabilities (such as Ethernet), it is not necessary to send hellos reliably to all neighbors individually. Therefore, EIGRP sends a single multicast hello with an indication in the packet informing the receivers that the packet need not be acknowledged. Other types of packets (such as updates) require acknowledgment, which is shown in the packet. The reliable transport has a provision to send multicast packets quickly when there are unacknowledged packets pending. Doing so helps ensure that convergence time remains low in the presence of varying speed links.
- The DUAL finite state machine embodies the decision process for all route computations. It tracks all routes advertised by all neighbors. DUAL uses the distance information (known as a metric) to select efficient, loop-free paths. DUAL selects routes to be inserted into a routing table based on feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination that is guaranteed not to be part of a routing loop. When there are no feasible successors, but there are neighbors advertising the destination, a recomputation must occur. This is the process whereby a new successor is determined. The amount of time it takes to recompute the route affects the convergence time. Recomputation is processor-intensive; it is advantageous to avoid recomputation if it is not necessary. When a topology change occurs, DUAL tests for feasible successors. If there are feasible successors, it uses any it finds to avoid unnecessary recomputation.
- The protocol-dependent modules are responsible for network layer protocol-specific tasks. An example is the IP EIGRP module, which is responsible for sending and receiving EIGRP packets that are encapsulated in IP. It is also responsible for parsing EIGRP packets and informing DUAL of the new information received. EIGRP asks DUAL to make routing decisions, but the results are stored in the IP routing table. EIGRP is also responsible for redistributing routes learned by other IP routing protocols.



Note To enable EIGRP, the Device or active switch must be running the

How to Configure EIGRP

To create an EIGRP routing process, you must enable EIGRP and associate networks. EIGRP sends updates to the interfaces in the specified networks. If you do not specify an interface network, it is not advertised in any EIGRP update.



Note If you have routers on your network that are configured for IGRP, and you want to change to EIGRP, you must designate transition routers that have both IGRP and EIGRP configured. In these cases, perform Steps 1 through 3 in the next section and also see the “Configuring Split Horizon” section. You must use the same AS number for routes to be automatically redistributed.

Default EIGRP Configuration

Table 107: Default EIGRP Configuration

Feature	Default Setting
Auto summary	Disabled.
Default-information	Exterior routes are accepted and default information is passed between processes when doing redistribution.
Default metric	Only connected routes and interface static routes can be redistributed with the default metric. The metric includes: <ul style="list-style-type: none"> • Bandwidth: 0 or greater kb/s. • Delay (tens of microseconds): 0 or any positive number that is less than 39.1 nanoseconds. • Reliability: any number between 0 and 255 (255 means 100 percent reliability). • Loading: effective bandwidth as a number between 0 and 255 (255 means 100 percent loading). • MTU: maximum transmission unit size of the route in bytes. 0 or any positive integer.
Distance	Internal distance: 90. External distance: 170.
EIGRP log-neighbor changes	Disabled. No adjacency changes logged.
IP authentication key-chain	No authentication provided.
IP authentication mode	No authentication provided.
IP bandwidth-percent	50 percent.

Feature	Default Setting
IP hello interval	For low-speed nonbroadcast multiaccess (NBMA) networks: 60 seconds; for other networks: 5 seconds.
IP hold-time	For low-speed NBMA networks: 180 seconds; all other networks: 15 seconds.
IP split-horizon	Enabled.
IP summary address	No summary aggregate addresses are predefined.
Metric weights	tos: 0; k1 and k3: 1; k2, k4, and k5: 0
Network	None specified.
Nonstop Forwarding (NSF) Awareness	Enabled for IPv4 on switches running the Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware and software changes.
NSF capability	Disabled. Note The Device supports EIGRP NSF-capable routing for IPv4.
Offset-list	Disabled.
Router EIGRP	Disabled.
Set metric	No metric set in the route map.
Traffic-share	Distributed proportionately to the ratios of the metrics.
Variance	1 (equal-cost load-balancing).

EIGRP Nonstop Forwarding

The Device stack supports two levels of EIGRP nonstop forwarding:

- EIGRP NSF Awareness
- EIGRP NSF Capability

EIGRP NSF Awareness

The Device supports EIGRP NSF Awareness for IPv4. When the neighboring router is NSF-capable, the Layer 3 Device continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router failing and the backup RP taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade.

This feature cannot be disabled. For more information on this feature, see the “EIGRP Nonstop Forwarding (NSF) Awareness” section of the *Cisco IOS IP Routing Protocols Configuration Guide, Release 12.4*.

EIGRP NSF Capability

The Device supports EIGRP Cisco NSF routing to speed up convergence and to eliminate traffic loss after a stack's active switch changeover.

The also supports EIGRP NSF-capable routing for IPv4 for better convergence and lower traffic loss following an active switch changeover. When an EIGRP NSF-capable active switch restarts or a new active switch starts up and NSF restarts, the Device has no neighbors, and the topology table is empty. The Device must bring up the interfaces, reacquire neighbors, and rebuild the topology and routing tables without interrupting the traffic directed toward the Device stack. EIGRP peer routers maintain the routes learned from the new active switch and continue forwarding traffic through the NSF restart process.

To prevent an adjacency reset by the neighbors, the new active switch uses a new Restart (RS) bit in the EIGRP packet header to show the restart. When the neighbor receives this, it synchronizes the stack in its peer list and maintains the adjacency with the stack. The neighbor then sends its topology table to the active switch with the RS bit set to show that it is NSF-aware and is aiding the new active switch.

If at least one of the stack peer neighbors is NSF-aware, the active switch receives updates and rebuilds its database. Each NSF-aware neighbor sends an end of table (EOT) marker in the last update packet to mark the end of the table content. The active switch recognizes the convergence when it receives the EOT marker, and it then begins sending updates. When the active switch has received all EOT markers from its neighbors or when the NSF converge timer expires, EIGRP notifies the routing information database (RIB) of convergence and floods its topology table to all NSF-aware peers.

Configuring Basic EIGRP Parameters

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	router eigrp autonomous-system Example: <pre>Switch(config)# router eigrp 10</pre>	Enables an EIGRP routing process, and enter router configuration mode. The AS number identifies the routes to other EIGRP routers and is used to tag routing information.
Step 3	nsf Example: <pre>Switch(config-router)# nsf</pre>	(Optional) Enables EIGRP NSF. Enter this command on the active switch and on all of its peers.
Step 4	network network-number Example: <pre>Switch(config-router)# network 192.168.0.0</pre>	Associate networks with an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks.
Step 5	eigrp log-neighbor-changes Example:	(Optional) Enables logging of EIGRP neighbor changes to monitor routing system stability.

	Command or Action	Purpose
	Switch(config-router)# eigrp log-neighbor-changes	
Step 6	metric weights <i>tos k1 k2 k3 k4 k5</i> Example: Switch(config-router)# metric weights 0 2 0 2 0 0	(Optional) Adjust the EIGRP metric. Although the defaults have been carefully set to provide excellent operation in most networks, you can adjust them. Caution Setting metrics is complex and is not recommended without guidance from an experienced network designer.
Step 7	offset-list [<i>access-list number name</i>] { in out } <i>offset</i> [<i>type number</i>] Example: Switch(config-router)# offset-list 21 out 10	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through EIGRP. You can limit the offset list with an access list or an interface.
Step 8	auto-summary Example: Switch(config-router)# auto-summary	(Optional) Enables automatic summarization of subnet routes into network-level routes.
Step 9	interface <i>interface-id</i> Example: Switch(config-router)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 10	ip summary-address eigrp <i>autonomous-system-number address mask</i> Example: Switch(config-if)# ip summary-address eigrp 1 192.168.0.0 255.255.0.0	(Optional) Configures a summary aggregate.
Step 11	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 12	show ip protocols Example: Switch# show ip protocols	Verifies your entries. For NSF awareness, the output shows: *** IP Routing is NSF aware *** EIGRP NSF enabled
Step 13	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Configuring EIGRP Interfaces

Other optional EIGRP parameters can be configured on an interface basis.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	ip bandwidth-percent eigrp <i>percent</i> Example: Switch(config-if)# <code>ip bandwidth-percent eigrp 60</code>	(Optional) Configures the percentage of bandwidth that can be used by EIGRP on an interface. The default is 50 percent.
Step 4	ip summary-address eigrp <i>autonomous-system-number address mask</i> Example: Switch(config-if)# <code>ip summary-address eigrp 109 192.161.0.0 255.255.0.0</code>	(Optional) Configures a summary aggregate address for a specified interface (not usually necessary if auto-summary is enabled).
Step 5	ip hello-interval eigrp <i>autonomous-system-number seconds</i> Example: Switch(config-if)# <code>ip hello-interval eigrp 109 10</code>	(Optional) Change the hello time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 60 seconds for low-speed NBMA networks and 5 seconds for all other networks.
Step 6	ip hold-time eigrp <i>autonomous-system-number seconds</i> Example: Switch(config-if)# <code>ip hold-time eigrp 109 40</code>	(Optional) Change the hold time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 180 seconds for low-speed NBMA networks and 15 seconds for all other networks. Caution Do not adjust the hold time without consulting Cisco technical support.

	Command or Action	Purpose
Step 7	no ip split-horizon eigrp <i>autonomous-system-number</i> Example: Switch(config-if)# no ip split-horizon eigrp 109	(Optional) Disables split horizon to allow route information to be advertised by a router out any interface from which that information originated.
Step 8	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 9	show ip eigrp interface Example: Switch# show ip eigrp interface	Displays which interfaces EIGRP is active on and information about EIGRP relating to those interfaces.
Step 10	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring EIGRP Route Authentication

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol to prevent the introduction of unauthorized or false routing messages from unapproved sources.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	ip authentication mode eigrp <i>autonomous-system</i> md5 Example: Switch(config-if)# ip authentication mode eigrp 104 md5	Enables MD5 authentication in IP EIGRP packets.

	Command or Action	Purpose
Step 4	<p>ip authentication key-chain eigrp <i>autonomous-system</i> <i>key-chain</i></p> <p>Example:</p> <pre>Switch(config-if)# ip authentication key-chain eigrp 105 chain1</pre>	Enables authentication of IP EIGRP packets.
Step 5	<p>exit</p> <p>Example:</p> <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	<p>key chain <i>name-of-chain</i></p> <p>Example:</p> <pre>Switch(config)# key chain chain1</pre>	Identify a key chain and enter key-chain configuration mode. Match the name configured in Step 4.
Step 7	<p>key <i>number</i></p> <p>Example:</p> <pre>Switch(config-keychain)# key 1</pre>	In key-chain configuration mode, identify the key number.
Step 8	<p>key-string <i>text</i></p> <p>Example:</p> <pre>Switch(config-keychain-key)# key-string key1</pre>	In key-chain key configuration mode, identify the key string.
Step 9	<p>accept-lifetime <i>start-time</i> {infinite <i>end-time</i> duration} <i>seconds</i>}</p> <p>Example:</p> <pre>Switch(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 2011 duration 7200</pre>	<p>(Optional) Specifies the time period during which the key can be received.</p> <p>The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month date year</i> or <i>hh:mm:ss date Month year</i>. The default is forever with the default <i>start-time</i> and the earliest acceptable date as January 1, 1993. The default <i>end-time</i> and duration is infinite.</p>
Step 10	<p>send-lifetime <i>start-time</i> {infinite <i>end-time</i> duration} <i>seconds</i>}</p> <p>Example:</p> <pre>Switch(config-keychain-key)# send-lifetime 14:00:00 Jan 25 2011 duration 3600</pre>	<p>(Optional) Specifies the time period during which the key can be sent.</p> <p>The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month date year</i> or <i>hh:mm:ss date Month year</i>. The default is forever with the default <i>start-time</i> and the earliest acceptable date as January 1, 1993. The default <i>end-time</i> and duration is infinite.</p>
Step 11	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 12	show key chain Example: <pre>Switch# show key chain</pre>	Displays authentication key information.
Step 13	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

EIGRP Stub Routing

The EIGRP stub routing feature reduces resource utilization by moving routed traffic closer to the end user.



Note The device uses EIGRP stub routing at the access layer to eliminate the need for other types of routing advertisements.

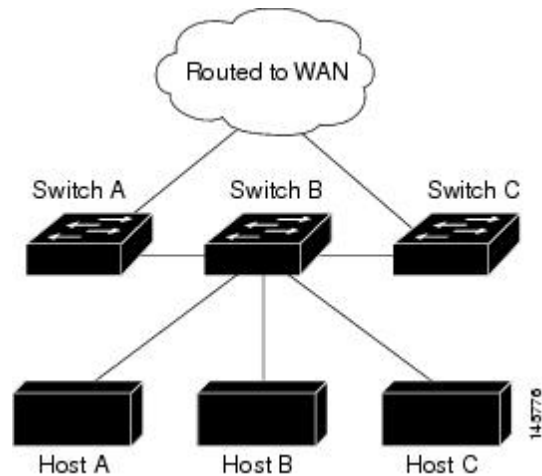
In a network using EIGRP stub routing, the only allowable route for IP traffic to the user is through a device that is configured with EIGRP stub routing. The device sends the routed traffic to interfaces that are configured as user interfaces or are connected to other devices.

When using EIGRP stub routing, you need to configure the distribution and remote routers to use EIGRP and to configure only the device as a stub. Only specified routes are propagated from the device. The device responds to all queries for summaries, connected routes, and routing updates.

Any neighbor that receives a packet informing it of the stub status does not query the stub router for any routes, and a router that has a stub peer does not query that peer. The stub router depends on the distribution router to send the proper updates to all peers.

In the figure given below, device B is configured as an EIGRP stub router. Devices A and C are connected to the rest of the WAN. Device B advertises connected, static, redistribution, and summary routes to Device A and C. Device B does not advertise any routes learned from Device A (and the reverse).

Figure 86: EIGRP Stub Router Configuration



Monitoring and Maintaining EIGRP

You can delete neighbors from the neighbor table. You can also display various EIGRP routing statistics. The table given below lists the privileged EXEC commands for deleting neighbors and displaying statistics.

Table 108: IP EIGRP Clear and Show Commands

clear ip eigrp neighbors [<i>if-address</i> <i>interface</i>]	Deletes neighbors from the neighbor table.
show ip eigrp interface [<i>interface</i>] [<i>as number</i>]	Displays information about interfaces participating in EIGRP.
show ip eigrp neighbors [<i>type-number</i>]	Displays EIGRP discovered neighbors.
show ip eigrp topology [<i>autonomous-system-number</i>] [[<i>ip-address</i>] <i>mask</i>]	Displays the EIGRP topology table.
show ip eigrp traffic [<i>autonomous-system-number</i>]	Displays the number of packets sent and received by the EIGRP process.

Information About Multi-VRF CE

Virtual Private Networks (VPNs) provide a secure way for customers to share bandwidth over an ISP backbone network. A VPN is a collection of sites sharing a common routing table. A customer site is connected to the service-provider network by one or more interfaces, and the service provider associates each interface with a VPN routing table, called a VPN routing/forwarding (VRF) table.

The switch supports multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE) when it is running the . Multi-VRF CE allows a service provider to support two or more VPNs with overlapping IP addresses.



Note The switch does not use Multiprotocol Label Switching (MPLS) to support VPNs.

Understanding Multi-VRF CE

Multi-VRF CE is a feature that allows a service provider to support two or more VPNs, where IP addresses can be overlapped among the VPNs. Multi-VRF CE uses input interfaces to distinguish routes for different VPNs and forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be either physical, such as Ethernet ports, or logical, such as VLAN SVIs, but an interface cannot belong to more than one VRF at any time.



Note Multi-VRF CE interfaces must be Layer 3 interfaces.

Multi-VRF CE includes these devices:

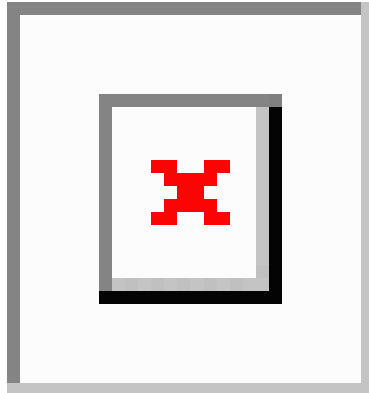
- Customer edge (CE) devices provide customers access to the service-provider network over a data link to one or more provider edge routers. The CE device advertises the site's local routes to the router and learns the remote VPN routes from it. A switch can be a CE.
- Provider edge (PE) routers exchange routing information with CE devices by using static routing or a routing protocol such as BGP, RIPv2, OSPF, or EIGRP. The PE is only required to maintain VPN routes for those VPNs to which it is directly attached, eliminating the need for the PE to maintain all of the service-provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites. Multiple interfaces on a PE router can be associated with a single VRF if all of these sites participate in the same VPN. Each VPN is mapped to a specified VRF. After learning local VPN routes from CEs, a PE router exchanges VPN routing information with other PE routers by using internal BGP (IBPG).
- Provider routers or core routers are any routers in the service provider network that do not attach to CE devices.

With multi-VRF CE, multiple customers can share one CE, and only one physical link is used between the CE and the PE. The shared CE maintains separate VRF tables for each customer and switches or routes packets for each customer based on its own routing table. Multi-VRF CE extends limited PE functionality to a CE device, giving it the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

Network Topology

The figure shows a configuration using switches as multiple virtual CEs. This scenario is suited for customers who have low bandwidth requirements for their VPN service, for example, small companies. In this case, multi-VRF CE support is required in the switches. Because multi-VRF CE is a Layer 3 feature, each interface in a VRF must be a Layer 3 interface.

Figure 87: Switches Acting as Multiple Virtual CEs



When the CE switch receives a command to add a Layer 3 interface to a VRF, it sets up the appropriate mapping between the VLAN ID and the policy label (PL) in multi-VRF-CE-related data structures and adds the VLAN ID and PL to the VLAN database.

When multi-VRF CE is configured, the Layer 3 forwarding table is conceptually partitioned into two sections:

- The multi-VRF CE routing section contains the routes from different VPNs.
- The global routing section contains routes to non-VPN networks, such as the Internet.

VLAN IDs from different VRFs are mapped into different policy labels, which are used to distinguish the VRFs during processing. For each new VPN route learned, the Layer 3 setup function retrieves the policy label by using the VLAN ID of the ingress port and inserts the policy label and new route to the multi-VRF CE routing section. If the packet is received from a routed port, the port internal VLAN ID number is used; if the packet is received from an SVI, the VLAN number is used.

Packet-Forwarding Process

This is the packet-forwarding process in a multi-VRF-CE-enabled network:

- When the switch receives a packet from a VPN, the switch looks up the routing table based on the input policy label number. When a route is found, the switch forwards the packet to the PE.
- When the ingress PE receives a packet from the CE, it performs a VRF lookup. When a route is found, the router adds a corresponding MPLS label to the packet and sends it to the MPLS network.
- When an egress PE receives a packet from the network, it strips the label and uses the label to identify the correct VPN routing table. Then it performs the normal route lookup. When a route is found, it forwards the packet to the correct adjacency.
- When a CE receives a packet from an egress PE, it uses the input policy label to look up the correct VPN routing table. If a route is found, it forwards the packet within the VPN.

Network Components

To configure VRF, you create a VRF table and specify the Layer 3 interface associated with the VRF. Then configure the routing protocols in the VPN and between the CE and the PE. BGP is the preferred routing protocol used to distribute VPN routing information across the provider's backbone. The multi-VRF CE network has three major components:

- VPN route target communities—lists of all other members of a VPN community. You need to configure VPN route targets for each VPN community member.
- Multiprotocol BGP peering of VPN community PE routers—propagates VRF reachability information to all members of a VPN community. You need to configure BGP peering in all PE routers within a VPN community.
- VPN forwarding—transports all traffic between all VPN community members across a VPN service-provider network.

VRF-Aware Services

IP services can be configured on global interfaces, and these services run within the global routing instance. IP services are enhanced to run on multiple routing instances; they are VRF-aware. Any configured VRF in the system can be specified for a VRF-aware service.

VRF-Aware services are implemented in platform-independent modules. VRF means multiple routing instances in Cisco IOS. Each platform has its own limit on the number of VRFs it supports.

VRF-aware services have the following characteristics:

- The user can ping a host in a user-specified VRF.
- ARP entries are learned in separate VRFs. The user can display Address Resolution Protocol (ARP) entries for specific VRFs.

How to Configure Multi-VRF CE

Default Multi-VRF CE Configuration

Table 109: Default VRF Configuration

Feature	Default Setting
VRF	Disabled. No VRFs are defined.
Maps	No import maps, export maps, or route maps are defined.
VRF maximum routes	Fast Ethernet switches: 8000 Gigabit Ethernet switches: 12000.
Forwarding table	The default for an interface is the global routing table.

Multi-VRF CE Configuration Guidelines



Note To use multi-VRF CE, you must have the enabled on your switch.

- A switch with multi-VRF CE is shared by multiple customers, and each customer has its own routing table.
- Because customers use different VRF tables, the same IP addresses can be reused. Overlapped IP addresses are allowed in different VPNs.
- Multi-VRF CE lets multiple customers share the same physical link between the PE and the CE. Trunk ports with multiple VLANs separate packets among customers. Each customer has its own VLAN.
- Multi-VRF CE does not support all MPLS-VRF functionality. It does not support label exchange, LDP adjacency, or labeled packets.
- For the PE router, there is no difference between using multi-VRF CE or using multiple CEs. In Figure 41-6, multiple virtual Layer 3 interfaces are connected to the multi-VRF CE device.
- The switch supports configuring VRF by using physical ports, VLAN SVIs, or a combination of both. The SVIs can be connected through an access port or a trunk port.
- A customer can use multiple VLANs as long as they do not overlap with those of other customers. A customer's VLANs are mapped to a specific routing table ID that is used to identify the appropriate routing tables stored on the switch.
- The switch supports one global network and up to 25 VRFs.
- Most routing protocols (BGP, OSPF, RIP, and static routing) can be used between the CE and the PE. However, we recommend using external BGP (EBGP) for these reasons:
 - BGP does not require multiple algorithms to communicate with multiple CEs.
 - BGP is designed for passing routing information between systems run by different administrations.
 - BGP makes it easy to pass attributes of the routes to the CE.
- Multi-VRF CE does not affect the packet switching rate.
- VPN multicast is not supported.
- You can enable VRF on a private VLAN, and the reverse.
- You cannot enable VRF when policy-based routing (PBR) is enabled on an interface, and the reverse.
- You cannot enable VRF when Web Cache Communication Protocol (WCCP) is enabled on an interface, and the reverse.

Configuring VRFs

Perform the following steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ip routing Example: Switch(config)# ip routing	Enables IP routing.
Step 3	ip vrf vrf-name Example: Switch(config)# ip vrf vpn1	Names the VRF, and enter VRF configuration mode.
Step 4	rd route-distinguisher Example: Switch(config-vrf)# rd 100:2	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y)
Step 5	route-target {export import both} <i>route-target-ext-community</i> Example: Switch(config-vrf)# route-target both 100:2	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The <i>route-target-ext-community</i> should be the same as the <i>route-distinguisher</i> entered in Step 4.
Step 6	import map route-map Example: Switch(config-vrf)# import map importmap1	(Optional) Associates a route map with the VRF.
Step 7	interface interface-id Example: Switch(config-vrf)# interface gigabitethernet 1/0/1	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or SVI.
Step 8	ip vrf forwarding vrf-name Example: Switch(config-if)# ip vrf forwarding vpn1	Associates the VRF with the Layer 3 interface. Note When ip vrf forwarding is enabled in the Management Interface, the access point does not join.
Step 9	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 10	show ip vrf [brief detail interfaces] [vrf-name] Example: Switch# show ip vrf interfaces vpn1	Verifies the configuration. Displays information about the configured VRFs.
Step 11	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring VRF-Aware Services

These services are VRF-Aware:

- ARP
- Ping
- Simple Network Management Protocol (SNMP)
- Unicast Reverse Path Forwarding (uRPF)
- Syslog
- Traceroute
- FTP and TFTP

Configuring VRF-Aware Services for ARP

Procedure

	Command or Action	Purpose
Step 1	show ip arp vrf vrf-name Example: Switch# show ip arp vrf vpn1	Displays the ARP table in the specified VRF.

Configuring VRF-Aware Services for Ping

Procedure

	Command or Action	Purpose
Step 1	<p>ping vrf <i>vrf-name</i> ip-host</p> <p>Example:</p> <pre>Switch# ping vrf vpn1 ip-host</pre>	Displays the ARP table in the specified VRF.

Configuring VRF-Aware Services for SNMP

Procedure

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>snmp-server trap authentication vrf</p> <p>Example:</p> <pre>Switch(config)# snmp-server trap authentication vrf</pre>	Enables SNMP traps for packets on a VRF.
Step 3	<p>snmp-server engineID remote <i>host</i> vrf <i>vpn-instance</i> <i>engine-id string</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server engineID remote 172.16.20.3 vrf vpn1 80000009030000B064EFE100</pre>	Configures a name for the remote SNMP engine on a switch.
Step 4	<p>snmp-server host <i>host</i> vrf <i>vpn-instance</i> traps <i>community</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server host 172.16.20.3 vrf vpn1 traps comaccess</pre>	Specifies the recipient of an SNMP trap operation and specifies the VRF table to be used for sending SNMP traps.
Step 5	<p>snmp-server host <i>host</i> vrf <i>vpn-instance</i> informs <i>community</i></p> <p>Example:</p> <pre>Switch(config)# snmp-server host 172.16.20.3 vrf vpn1 informs comaccess</pre>	Specifies the recipient of an SNMP inform operation and specifies the VRF table to be used for sending SNMP informs.

	Command or Action	Purpose
Step 6	snmp-server user <i>user group remote host vrf vpn-instance security model</i> Example: <pre>Switch(config)# snmp-server user abcd remote 172.16.20.3 vrf vpn1 priv v2c 3des secure3des</pre>	Adds a user to an SNMP group for a remote host on a VRF for SNMP access.
Step 7	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring VRF-Aware Services for HSRP

HSRP support for VRFs ensures that HSRP virtual IP addresses are added to the correct IP routing table.

For complete syntax and usage information for the commands, refer to the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	no switchport Example: <pre>Switch(config-if)# no switchport</pre>	Removes the interface from Layer 2 configuration mode if it is a physical interface.
Step 4	ip vrf forwarding <i>vrf-name</i> Example: <pre>Switch(config-if)# ip vrf forwarding vpn1</pre>	Configures VRF on the interface.
Step 5	ip address <i>ip-address</i> Example:	Enters the IP address for the interface.

	Command or Action	Purpose
	Switch(config-if)# ip address 10.1.5.1	
Step 6	standby 1 ip ip-address Example: Switch(config-if)#standby 1 ip 10.1.1.254	Enables HSRP and configure the virtual IP address.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Configuring VRF-Aware Services for uRPF

uRPF can be configured on an interface assigned to a VRF, and source lookup is done in the VRF table.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	no switchport Example: Switch(config-if)# no switchport	Removes the interface from Layer 2 configuration mode if it is a physical interface.
Step 4	ip vrf forwarding vrf-name Example: Switch(config-if)# ip vrf forwarding vpn2	Configures VRF on the interface.
Step 5	ip address ip-address Example: Switch(config-if)# ip address 10.1.5.1	Enters the IP address for the interface.

	Command or Action	Purpose
Step 6	ip verify unicast reverse-path Example: Switch(config-if)# ip verify unicast reverse-path	Enables uRPF on the interface.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Configuring VRF-Aware RADIUS

To configure VRF-Aware RADIUS, you must first enable AAA on a RADIUS server. The switch supports the **ip vrf forwarding** *vrf-name* server-group configuration and the **ip radius source-interface** global configuration commands, as described in the Per VRF AAA Feature Guide.

Configuring VRF-Aware Services for Syslog

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	logging on Example: Switch(config)# logging on	Enables or temporarily disables logging of storage router event message.
Step 3	logging host ip-address vrf vrf-name Example: Switch(config)# logging host 10.10.1.0 vrf vpn1	Specifies the host address of the syslog server where logging messages are to be sent.
Step 4	logging buffered logging buffered size debugging Example: Switch(config)# logging buffered critical 6000 debugging	Logs messages to an internal buffer.
Step 5	logging trap debugging Example:	Limits the logging messages sent to the syslog server.

	Command or Action	Purpose
	Switch(config)# logging trap debugging	
Step 6	logging facility <i>facility</i> Example: Switch(config)# logging facility user	Sends system logging messages to a logging facility.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Configuring VRF-Aware Services for Traceroute

Procedure

	Command or Action	Purpose
Step 1	traceroute vrf <i>vrf-name ipaddress</i> Example: Switch(config)# traceroute vrf vpn2 10.10.1.1	Specifies the name of a VPN VRF in which to find the destination address.

Configuring VRF-Aware Services for FTP and TFTP

So that FTP and TFTP are VRF-aware, you must configure some FTP/TFTP CLIs. For example, if you want to use a VRF table that is attached to an interface, say E1/0, you need to configure the `ip tftp source-interface E1/0` or the `ip ftp source-interface E1/0` command to inform TFTP or FTP server to use a specific routing table. In this example, the VRF table is used to look up the destination IP address. These changes are backward-compatible and do not affect existing behavior. That is, you can use the source-interface CLI to send packets out a particular interface even if no VRF is configured on that interface.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ip ftp source-interface <i>interface-type interface-number</i> Example: Switch(config)# ip ftp source-interface gigabitethernet 1/0/2	Specifies the source IP address for FTP connections.

	Command or Action	Purpose
Step 3	end Example: Switch(config)#end	Returns to privileged EXEC mode.
Step 4	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 5	ip tftp source-interface <i>interface-type interface-number</i> Example: Switch(config)# ip tftp source-interface gigabitethernet 1/0/2	Specifies the source IP address for TFTP connections.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Configuring Multicast VRFs

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ip routing Example: Switch(config)# ip routing	Enables IP routing mode.
Step 3	ip vrf <i>vrf-name</i> Example: Switch(config)# ip vrf vpn1	Names the VRF, and enter VRF configuration mode.

	Command or Action	Purpose
Step 4	rd <i>route-distinguisher</i> Example: <pre>Switch(config-vrf)# rd 100:2</pre>	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y)
Step 5	route-target { export import both } <i>route-target-ext-community</i> Example: <pre>Switch(config-vrf)# route-target import 100:2</pre>	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The <i>route-target-ext-community</i> should be the same as the <i>route-distinguisher</i> entered in Step 4.
Step 6	import map <i>route-map</i> Example: <pre>Switch(config-vrf)# import map importmap1</pre>	(Optional) Associates a route map with the VRF.
Step 7	ip multicast-routing vrf <i>vrf-name</i> distributed Example: <pre>Switch(config-vrf)# ip multicast-routing vrf vpn1 distributed</pre>	(Optional) Enables global multicast routing for VRF table.
Step 8	interface <i>interface-id</i> Example: <pre>Switch(config-vrf)# interface gigabitethernet 1/0/2</pre>	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or an SVI.
Step 9	ip vrf forwarding <i>vrf-name</i> Example: <pre>Switch(config-if)# ip vrf forwarding vpn1</pre>	Associates the VRF with the Layer 3 interface.
Step 10	ip address <i>ip-address</i> mask Example: <pre>Switch(config-if)# ip address 10.1.5.1 255.255.255.0</pre>	Configures IP address for the Layer 3 interface.
Step 11	ip pim sparse-dense mode Example: <pre>Switch(config-if)# ip pim sparse-dense mode</pre>	Enables PIM on the VRF-associated Layer 3 interface.
Step 12	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 13	show ip vrf [brief detail interfaces] [vrf-name] Example: Switch# show ip vrf detail vpn1	Verifies the configuration. Displays information about the configured VRFs.
Step 14	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a VPN Routing Session

Routing within the VPN can be configured with any supported routing protocol (RIP, OSPF, EIGRP, or BGP) or with static routing. The configuration shown here is for OSPF, but the process is the same for other protocols.



Note To configure an EIGRP routing process to run within a VRF instance, you must configure an autonomous-system number by entering the **autonomous-system** *autonomous-system-number* address-family configuration mode command.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router ospf process-id vrf vrf-name Example: Switch(config)# router ospf 1 vrf vpn1	Enables OSPF routing, specifies a VPN forwarding table, and enter router configuration mode.
Step 3	log-adjacency-changes Example: Switch(config-router)# log-adjacency-changes	(Optional) Logs changes in the adjacency state. This is the default state.
Step 4	redistribute bgp autonomous-system-number subnets Example:	Sets the switch to redistribute information from the BGP network to the OSPF network.

	Command or Action	Purpose
	Switch(config-router)# redistribute bgp 10 subnets	
Step 5	network <i>network-number</i> area <i>area-id</i> Example: Switch(config-router)# network 1 area 2	Defines a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	end Example: Switch(config-router)# end	Returns to privileged EXEC mode.
Step 7	show ip ospf <i>process-id</i> Example: Switch# show ip ospf 1	Verifies the configuration of the OSPF network.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring BGP PE to CE Routing Sessions

Procedure

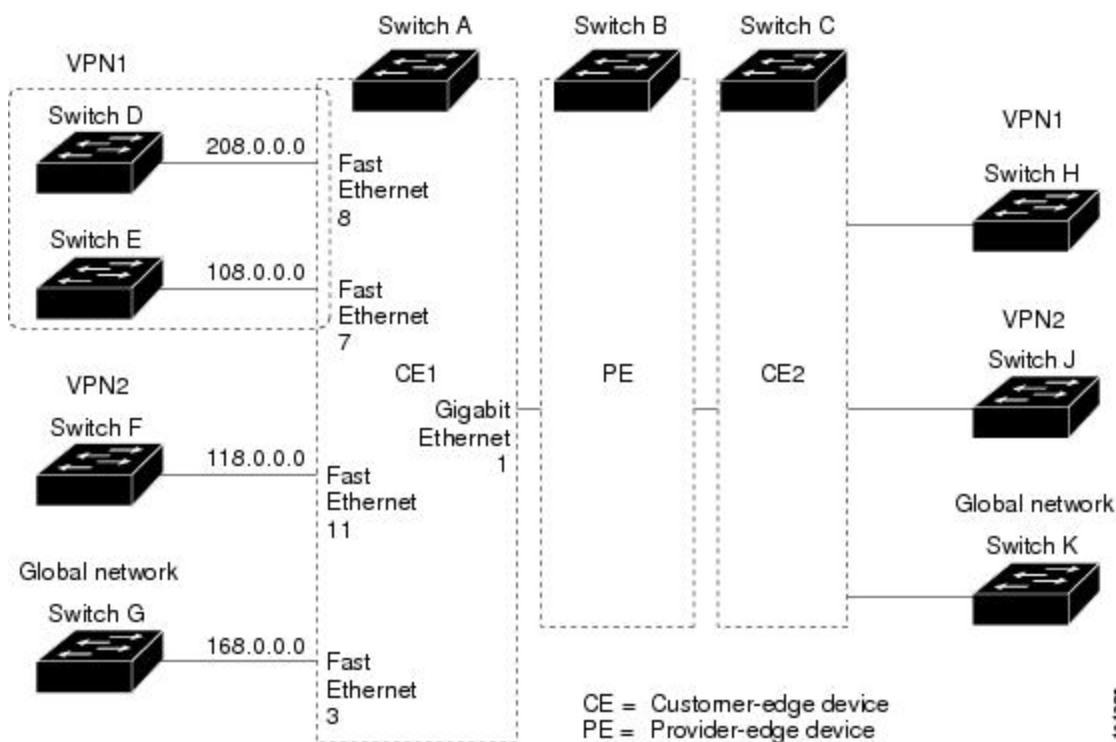
	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router bgp <i>autonomous-system-number</i> Example: Switch(config)# router bgp 2	Configures the BGP routing process with the AS number passed to other BGP routers, and enter router configuration mode.
Step 3	network <i>network-number</i> mask <i>network-mask</i> Example: Switch(config-router)# network 5 mask 255.255.255.0	Specifies a network and mask to announce using BGP.

	Command or Action	Purpose
Step 4	redistribute ospf <i>process-id</i> match internal Example: <pre>Switch(config-router)# redistribute ospf 1 match internal</pre>	Sets the switch to redistribute OSPF internal routes.
Step 5	network <i>network-number</i> area <i>area-id</i> Example: <pre>Switch(config-router)# network 5 area 2</pre>	Defines a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	address-family ipv4 vrf <i>vrf-name</i> Example: <pre>Switch(config-router)# address-family ipv4 vrf vpn1</pre>	Defines BGP parameters for PE to CE routing sessions, and enter VRF address-family mode.
Step 7	neighbor <i>address</i> remote-as <i>as-number</i> Example: <pre>Switch(config-router)# neighbor 10.1.1.2 remote-as 2</pre>	Defines a BGP session between PE and CE routers.
Step 8	neighbor <i>address</i> activate Example: <pre>Switch(config-router)# neighbor 10.2.1.1 activate</pre>	Activates the advertisement of the IPv4 address family.
Step 9	end Example: <pre>Switch(config-router)# end</pre>	Returns to privileged EXEC mode.
Step 10	show ip bgp [<i>ipv4</i>] [<i>neighbors</i>] Example: <pre>Switch# show ip bgp ipv4 neighbors</pre>	Verifies BGP configuration.
Step 11	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Multi-VRF CE Configuration Example

OSPF is the protocol used in VPN1, VPN2, and the global network. BGP is used in the CE to PE connections. The examples following the illustration show how to configure a switch as CE Switch A, and the VRF configuration for customer switches D and F. Commands for configuring CE Switch C and the other customer switches are not included but would be similar.

Figure 88: Multi-VRF CE Configuration Example



On Switch A, enable routing and configure VRF.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# ip vrf v11
Switch(config-vrf)# rd 800:1
Switch(config-vrf)# route-target export 800:1
Switch(config-vrf)# route-target import 800:1
Switch(config-vrf)# exit
Switch(config)# ip vrf v12
Switch(config-vrf)# rd 800:2
Switch(config-vrf)# route-target export 800:2
Switch(config-vrf)# route-target import 800:2
Switch(config-vrf)# exit
```

Configure the loopback and physical interfaces on Switch A. Gigabit Ethernet port 1 is a trunk connection to the PE. Gigabit Ethernet ports 8 and 11 connect to VPNs:

```
Switch(config)# interface loopback1
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 8.8.1.8 255.255.255.0
Switch(config-if)# exit
```



```
Switch(config)# interface loopback2
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 8.8.2.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface gigabitethernet1/0/5
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/8
Switch(config-if)# switchport access vlan 208
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/11
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit
```

Configure the VLANs used on Switch A. VLAN 10 is used by VRF 11 between the CE and the PE. VLAN 20 is used by VRF 12 between the CE and the PE. VLANs 118 and 208 are used for the VPNs that include Switch F and Switch D, respectively:

```
Switch(config)# interface vlan10
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 38.0.0.8 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface vlan20
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 83.0.0.8 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface vlan118
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 118.0.0.8 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface vlan208
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 208.0.0.8 255.255.255.0
Switch(config-if)# exit
```

Configure OSPF routing in VPN1 and VPN2.

```
Switch(config)# router ospf 1 vrf v11
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
Switch(config)# router ospf 2 vrf v12
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
```

Configure BGP for CE to PE routing.

```
Switch(config)# router bgp 800
Switch(config-router)# address-family ipv4 vrf v12
Switch(config-router-af)# redistribute ospf 2 match internal
Switch(config-router-af)# neighbor 83.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 83.0.0.3 activate
Switch(config-router-af)# network 8.8.2.0 mask 255.255.255.0
Switch(config-router-af)# exit
```

```
Switch(config-router)# address-family ipv4 vrf v11
Switch(config-router-af)# redistribute ospf 1 match internal
Switch(config-router-af)# neighbor 38.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 38.0.0.3 activate
Switch(config-router-af)# network 8.8.1.0 mask 255.255.255.0
Switch(config-router-af)# end
```

Switch D belongs to VPN 1. Configure the connection to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 208.0.0.20 255.255.255.0
Switch(config-if)# exit

Switch(config)# router ospf 101
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

Switch F belongs to VPN 2. Configure the connection to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface vlan118
Switch(config-if)# ip address 118.0.0.11 255.255.255.0
Switch(config-if)# exit

Switch(config)# router ospf 101
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

When used on switch B (the PE router), these commands configure only the connections to the CE device, Switch A.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip vrf v1
Router(config-vrf)# rd 100:1
Router(config-vrf)# route-target export 100:1
Router(config-vrf)# route-target import 100:1
Router(config-vrf)# exit

Router(config)# ip vrf v2
Router(config-vrf)# rd 100:2
Router(config-vrf)# route-target export 100:2
Router(config-vrf)# route-target import 100:2
Router(config-vrf)# exit
Router(config)# ip cef
Router(config)# interface Loopback1
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 3.3.1.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface Loopback2
```

```

Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 3.3.2.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.10
Router(config-if)# encapsulation dot1q 10
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 38.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.20
Router(config-if)# encapsulation dot1q 20
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 83.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# router bgp 100
Router(config-router)# address-family ipv4 vrf v2
Router(config-router-af)# neighbor 83.0.0.8 remote-as 800
Router(config-router-af)# neighbor 83.0.0.8 activate
Router(config-router-af)# network 3.3.2.0 mask 255.255.255.0
Router(config-router-af)# exit
Router(config-router)# address-family ipv4 vrf v1
Router(config-router-af)# neighbor 38.0.0.8 remote-as 800
Router(config-router-af)# neighbor 38.0.0.8 activate
Router(config-router-af)# network 3.3.1.0 mask 255.255.255.0
Router(config-router-af)# end

```

Monitoring Multi-VRF CE

Table 110: Commands for Displaying Multi-VRF CE Information

show ip protocols vrf <i>vrf-name</i>	Displays routing protocol information as a VRF.
show ip route vrf <i>vrf-name</i> [connected] [<i>protocol</i> [<i>as-number</i>]] [list] [mobile] [odr] [profile] [static] [summary] [supernets-only]	Displays IP routing table information as a VRF.
show ip vrf [brief detail interfaces] [<i>vrf-name</i>]	Displays information about the defined VRFs.

Configuring Unicast Reverse Path Forwarding

The unicast reverse path forwarding (unicast RPF) feature helps to mitigate problems that are caused by the introduction of malformed or forged (spoofed) IP source addresses into a network by discarding IP packets that lack a verifiable IP source address. For example, a number of common types of denial-of-service (DoS) attacks, including Smurf and Tribal Flood Network (TFN), can take advantage of forged or rapidly changing source IP addresses to allow attackers to thwart efforts to locate or filter the attacks. For Internet service providers (ISPs) that provide public access, Unicast RPF deflects such attacks by forwarding only packets that have source addresses that are valid and consistent with the IP routing table. This action protects the network of the ISP, its customer, and the rest of the Internet.



Note

- Unicast RPF is supported in .

Protocol-Independent Features

This section describes IP routing protocol-independent features that are available on switches running the feature set .

Distributed Cisco Express Forwarding

Information About Cisco Express Forwarding

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology used to optimize network performance. CEF implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. CEF is less CPU-intensive than fast switching route caching, allowing more CPU processing power to be dedicated to packet forwarding. In a switch stack, the hardware uses distributed CEF (dCEF) in the stack. In dynamic networks, fast switching cache entries are frequently invalidated because of routing changes, which can cause traffic to be process switched using the routing table, instead of fast switched using the route cache. CEF and dCEF use the Forwarding Information Base (FIB) lookup table to perform destination-based switching of IP packets.

The two main components in CEF and dCEF are the distributed FIB and the distributed adjacency tables.

- The FIB is similar to a routing table or information base and maintains a mirror image of the forwarding information in the IP routing table. When routing or topology changes occur in the network, the IP routing table is updated, and those changes are reflected in the FIB. The FIB maintains next-hop address information based on the information in the IP routing table. Because the FIB contains all known routes that exist in the routing table, CEF eliminates route cache maintenance, is more efficient for switching traffic, and is not affected by traffic patterns.
- Nodes in the network are said to be adjacent if they can reach each other with a single hop across a link layer. CEF uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all FIB entries.

Because the switch or switch stack uses Application Specific Integrated Circuits (ASICs) to achieve Gigabit-speed line rate IP traffic, CEF or dCEF forwarding applies only to the software-forwarding path, that is, traffic that is forwarded by the CPU.

How to Configure Cisco Express Forwarding

CEF or distributed CEF is enabled globally by default. If for some reason it is disabled, you can re-enable it by using the **ip cef** or **ip cef distributed** global configuration command.

The default configuration is CEF or dCEF enabled on all Layer 3 interfaces. Entering the **no ip route-cache cef** interface configuration command disables CEF for traffic that is being forwarded by software. This command does not affect the hardware forwarding path. Disabling CEF and using the **debug ip packet detail** privileged EXEC command can be useful to debug software-forwarded traffic. To enable CEF on an interface for the software-forwarding path, use the **ip route-cache cef** interface configuration command.



Caution Although the **no ip route-cache cef** interface configuration command to disable CEF on an interface is visible in the CLI, we strongly recommend that you do not disable CEF or dCEF on interfaces except for debugging purposes.

To enable CEF or dCEF globally and on an interface for software-forwarded traffic if it has been disabled:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	ip cef Example: Switch(config)# <code>ip cef</code>	Enables CEF operation on a non-stacking switch. Go to Step 4.
Step 3	ip cef distributed Example: Switch(config)# <code>ip cef distributed</code>	Enables CEF operation on a active switch.
Step 4	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 5	ip route-cache cef Example: Switch(config-if)# <code>ip route-cache cef</code>	Enables CEF on the interface for software-forwarded traffic.
Step 6	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 7	show ip cef Example: Switch# <code>show ip cef</code>	Displays the CEF status on all interfaces.
Step 8	show cef linecard [detail] Example: Switch# <code>show cef linecard detail</code>	(Optional) Displays CEF-related interface information on a non-stacking switch.

	Command or Action	Purpose
Step 9	show cef linecard [<i>slot-number</i>] [<i>detail</i>] Example: Switch# show cef linecard 5 detail	(Optional) Displays CEF-related interface information on a switch by stack member for all switches in the stack or for the specified switch. (Optional) For <i>slot-number</i> , enter the stack member switch number.
Step 10	show cef interface [<i>interface-id</i>] Example: Switch# show cef interface gigabitethernet 1/0/1	Displays detailed CEF information for all interfaces or the specified interface.
Step 11	show adjacency Example: Switch# show adjacency	Displays CEF adjacency table information.
Step 12	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Number of Equal-Cost Routing Paths

Information About Equal-Cost Routing Paths

When a router has two or more routes to the same network with the same metrics, these routes can be thought of as having an equal cost. The term parallel path is another way to see occurrences of equal-cost routes in a routing table. If a router has two or more equal-cost paths to a network, it can use them concurrently. Parallel paths provide redundancy in case of a circuit failure and also enable a router to load balance packets over the available paths for more efficient use of available bandwidth. Equal-cost routes are supported across switches in a stack.

Even though the router automatically learns about and configures equal-cost routes, you can control the maximum number of parallel paths supported by an IP routing protocol in its routing table. Although the switch software allows a maximum of 32 equal-cost routes, the switch hardware will never use more than 16 paths per route.

How to Configure Equal-Cost Routing Paths

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	router {rip ospf eigrp} Example: Switch(config)# router eigrp	Enters router configuration mode.
Step 3	maximum-paths <i>maximum</i> Example: Switch(config-router)# maximum-paths 2	Sets the maximum number of parallel paths for the protocol routing table. The range is from 1 to 16; the default is 4 for most IP routing protocols, but only 1 for BGP.
Step 4	end Example: Switch(config-router)# end	Returns to privileged EXEC mode.
Step 5	show ip protocols Example: Switch# show ip protocols	Verifies the setting in the <i>Maximum path</i> field.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Static Unicast Routes

Information About Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

The switch retains static routes until you remove them. However, you can override static routes with dynamic routing information by assigning administrative distance values. Each dynamic routing protocol has a default administrative distance, as listed in Table 41-16. If you want a static route to be overridden by information from a dynamic routing protocol, set the administrative distance of the static route higher than that of the dynamic protocol.

Table 111: Dynamic Routing Protocol Default Administrative Distances

Route Source	Default Distance
Connected interface	0
Static route	1
Enhanced IRGP summary route	5

Route Source	Default Distance
Internal Enhanced IGRP	90
IGRP	100
OSPF	110
Unknown	225

Static routes that point to an interface are advertised through RIP, IGRP, and other dynamic routing protocols, whether or not static **redistribute** router configuration commands were specified for those routing protocols. These static routes are advertised because static routes that point to an interface are considered in the routing table to be connected and hence lose their static nature. However, if you define a static route to an interface that is not one of the networks defined in a network command, no dynamic routing protocols advertise the route unless a **redistribute** static command is specified for these protocols.

When an interface goes down, all static routes through that interface are removed from the IP routing table. When the software can no longer find a valid next hop for the address specified as the forwarding router's address in a static route, the static route is also removed from the IP routing table.

Configuring Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

Follow these steps to configure a static route:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip route prefix mask {address interface} [distance] Example: <pre>Device(config)# ip route prefix mask gigabitethernet 1/0/4</pre>	Establish a static route.
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 5	show ip route Example: Switch# show ip route	Displays the current state of the routing table to verify the configuration.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no ip route** *prefix mask {address| interface}* global configuration command to remove a static route. The device retains static routes until you remove them.

Default Routes and Networks

Information About Default Routes and Networks

A router might not be able to learn the routes to all other networks. To provide complete routing capability, you can use some routers as smart routers and give the remaining routers default routes to the smart router. (Smart routers have routing table information for the entire internetwork.) These default routes can be dynamically learned or can be configured in the individual routers. Most dynamic interior routing protocols include a mechanism for causing a smart router to generate dynamic default information that is then forwarded to other routers.

If a router has a directly connected interface to the specified default network, the dynamic routing protocols running on that device generate a default route. In RIP, it advertises the pseudonetwork 0.0.0.0.

A router that is generating the default for a network also might need a default of its own. One way a router can generate its own default is to specify a static route to the network 0.0.0.0 through the appropriate device.

When default information is passed through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. In IGRP networks, there might be several candidate networks for the system default. Cisco routers use administrative distance and metric information to set the default route or the gateway of last resort.

If dynamic default information is not being passed to the system, candidates for the default route are specified with the **ip default-network** global configuration command. If this network appears in the routing table from any source, it is flagged as a possible choice for the default route. If the router has no interface on the default network, but does have a path to it, the network is considered as a possible candidate, and the gateway to the best default path becomes the gateway of last resort.

How to Configure Default Routes and Networks

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	ip default-network <i>network number</i> Example: Switch(config)# <code>ip default-network 1</code>	Specifies a default network.
Step 3	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	show ip route Example: Switch# <code>show ip route</code>	Displays the selected default route in the gateway of last resort display.
Step 5	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Route Maps to Redistribute Routing Information

Information About Route Maps

The switch can run multiple routing protocols simultaneously, and it can redistribute information from one routing protocol to another. Redistributing information from one routing protocol to another applies to all supported IP-based routing protocols.

You can also conditionally control the redistribution of routes between routing domains by defining enhanced packet filters or route maps between the two domains. The **match** and **set** route-map configuration commands define the condition portion of a route map. The **match** command specifies that a criterion must be matched. The **set** command specifies an action to be taken if the routing update meets the conditions defined by the match command. Although redistribution is a protocol-independent feature, some of the **match** and **set** route-map configuration commands are specific to a particular protocol.

One or more **match** commands and one or more **set** commands follow a **route-map** command. If there are no **match** commands, everything matches. If there are no **set** commands, nothing is done, other than the match. Therefore, you need at least one **match** or **set** command.



Note A route map with no **set** route-map configuration commands is sent to the CPU, which causes high CPU utilization.

You can also identify route-map statements as **permit** or **deny**. If the statement is marked as a deny, the packets meeting the match criteria are sent back through the normal forwarding channels (destination-based routing). If the statement is marked as permit, set clauses are applied to packets meeting the match criteria. Packets that do not meet the match criteria are forwarded through the normal routing channel.

How to Configure a Route Map

Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.



Note The keywords are the same as defined in the procedure to control the route distribution.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	route-map <i>map-tag</i> [permit deny] [<i>sequence number</i>] Example: <pre>Switch(config)# route-map rip-to-ospf permit 4</pre>	Defines any route maps used to control redistribution and enter route-map configuration mode. <i>map-tag</i> —A meaningful name for the route map. The redistribute router configuration command uses this name to reference this route map. Multiple route maps might share the same map tag name. (Optional) If permit is specified and the match criteria are met for this route map, the route is redistributed as controlled by the set actions. If deny is specified, the route is not redistributed. <i>sequence number</i> (Optional)— Number that indicates the position a new route map is to have in the list of route maps already configured with the same name.
Step 3	match as-path <i>path-list-number</i> Example: <pre>Switch(config-route-map)#match as-path 10</pre>	Matches a BGP AS path access list.
Step 4	match community-list <i>community-list-number</i> [exact] Example:	Matches a BGP community list.

	Command or Action	Purpose
	Switch(config-route-map)# match community-list 150	
Step 5	match ip address { <i>access-list-number</i> <i>access-list-name</i> } [... <i>access-list-number</i> ... <i>access-list-name</i>] Example: Switch(config-route-map)# match ip address 5 80	Matches a standard access list by specifying the name or number. It can be an integer from 1 to 199.
Step 6	match metric metric-value Example: Switch(config-route-map)# match metric 2000	Matches the specified route metric. The <i>metric-value</i> can be an EIGRP metric with a specified value from 0 to 4294967295.
Step 7	match ip next-hop { <i>access-list-number</i> <i>access-list-name</i> } [... <i>access-list-number</i> ... <i>access-list-name</i>] Example: Switch(config-route-map)# match ip next-hop 8 45	Matches a next-hop router address passed by one of the access lists specified (numbered from 1 to 199).
Step 8	match tag tag value [... <i>tag-value</i>] Example: Switch(config-route-map)# match tag 3500	Matches the specified tag value in a list of one or more route tag values. Each can be an integer from 0 to 4294967295.
Step 9	match interface type number [... <i>type-number</i>] Example: Switch(config-route-map)# match interface gigabitethernet 1/0/1	Matches the specified next hop route out one of the specified interfaces.
Step 10	match ip route-source { <i>access-list-number</i> <i>access-list-name</i> } [... <i>access-list-number</i> ... <i>access-list-name</i>] Example: Switch(config-route-map)# match ip route-source 10 30	Matches the address specified by the specified advertised access lists.
Step 11	match route-type { local internal external [type-1 type-2]} Example: Switch(config-route-map)# match route-type local	Matches the specified route-type : <ul style="list-style-type: none"> • local—Locally generated BGP routes. • internal—OSPF intra-area and interarea routes or EIGRP internal routes. • external—OSPF external routes (Type 1 or Type 2) or EIGRP external routes.

	Command or Action	Purpose
Step 12	<p>set dampening <i>halflife reuse suppress max-suppress-time</i></p> <p>Example:</p> <pre>Switch(config-route-map)# set dampening 30 1500 10000 120</pre>	Sets BGP route dampening factors.
Step 13	<p>set local-preference <i>value</i></p> <p>Example:</p> <pre>Switch(config-route-map)# set local-preference 100</pre>	Assigns a value to a local BGP path.
Step 14	<p>set origin {igp egp as incomplete}</p> <p>Example:</p> <pre>Switch(config-route-map)#set origin igp</pre>	Sets the BGP origin code.
Step 15	<p>set as-path {tag prepend as-path-string}</p> <p>Example:</p> <pre>Switch(config-route-map)# set as-path tag</pre>	Modifies the BGP autonomous system path.
Step 16	<p>set level {level-1 level-2 level-1-2 stub-area backbone}</p> <p>Example:</p> <pre>Switch(config-route-map)# set level level-1-2</pre>	Sets the level for routes that are advertised into the specified area of the routing domain. The stub-area and backbone are OSPF NSSA and backbone areas.
Step 17	<p>set metric <i>metric value</i></p> <p>Example:</p> <pre>Switch(config-route-map)# set metric 100</pre>	Sets the metric value to give the redistributed routes (for EIGRP only). The <i>metric value</i> is an integer from -294967295 to 294967295.
Step 18	<p>set metric<i>bandwidth delay reliability loading mtu</i></p> <p>Example:</p> <pre>Switch(config-route-map)# set metric 10000 10 255 1 1500</pre>	<p>Sets the metric value to give the redistributed routes (for EIGRP only):</p> <ul style="list-style-type: none"> <i>bandwidth</i>—Metric value or IGRP bandwidth of the route in kilobits per second in the range 0 to 4294967295 <i>delay</i>—Route delay in tens of microseconds in the range 0 to 4294967295. <i>reliability</i>—Likelihood of successful packet transmission expressed as a number between 0 and 255, where 255 means 100 percent reliability and 0 means no reliability.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>loading</i>—Effective bandwidth of the route expressed as a number from 0 to 255 (255 is 100 percent loading). • <i>mtu</i>—Minimum maximum transmission unit (MTU) size of the route in bytes in the range 0 to 4294967295.
Step 19	set metric-type {type-1 type-2} Example: <pre>Switch(config-route-map)# set metric-type type-2</pre>	Sets the OSPF external metric type for redistributed routes.
Step 20	set metric-type internal Example: <pre>Switch(config-route-map)# set metric-type internal</pre>	Sets the multi-exit discriminator (MED) value on prefixes advertised to external BGP neighbor to match the IGP metric of the next hop.
Step 21	set weight <i>number</i> Example: <pre>Switch(config-route-map)# set weight 100</pre>	Sets the BGP weight for the routing table. The value can be from 1 to 65535.
Step 22	end Example: <pre>Switch(config-route-map)# end</pre>	Returns to privileged EXEC mode.
Step 23	show route-map Example: <pre>Switch# show route-map</pre>	Displays all route maps configured or only the one specified to verify configuration.
Step 24	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

How to Control Route Distribution

Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.



Note The keywords are the same as defined in the procedure to configure the route map for redistribution.

The metrics of one routing protocol do not necessarily translate into the metrics of another. For example, the RIP metric is a hop count, and the IGRP metric is a combination of five qualities. In these situations, an artificial metric is assigned to the redistributed route. Uncontrolled exchanging of routing information between different routing protocols can create routing loops and seriously degrade network operation.

If you have not defined a default redistribution metric that replaces metric conversion, some automatic metric translations occur between routing protocols:

- RIP can automatically redistribute static routes. It assigns static routes a metric of 1 (directly connected).
- Any protocol can redistribute other routing protocols if a default mode is in effect.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router { rip ospf eigrp } Example: Switch(config)# router eigrp 10	Enters router configuration mode.
Step 3	redistribute protocol [process-id] {level-1 level-1-2 level-2} [metric metric-value] [metric-type type-value] [match internal external type-value] [tag tag-value] [route-map map-tag] [weight weight] [subnets] Example: Switch(config-router)# redistribute eigrp 1	Redistributes routes from one routing protocol to another routing protocol. If no route-maps are specified, all routes are redistributed. If the keyword route-map is specified with no <i>map-tag</i> , no routes are distributed.
Step 4	default-metric number Example: Switch(config-router)# default-metric 1024	Cause the current routing protocol to use the same metric value for all redistributed routes (RIP and OSPF).
Step 5	default-metric bandwidth delay reliability loading mtu Example: Switch(config-router)# default-metric 1000 100 250 100 1500	Cause the EIGRP routing protocol to use the same metric value for all non-EIGRP redistributed routes.
Step 6	end Example: Switch(config-router)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show route-map Example: Switch# show route-map	Displays all route maps configured or only the one specified to verify configuration.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Policy-Based Routing

Information About Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can specify and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

With PBR, you classify traffic using access control lists (ACLs) and then make traffic go through a different path. PBR is applied to incoming packets. All packets received on an interface with PBR enabled are passed through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop.

- Route map statement marked as permit is processed as follows:
 - A match command can match on length or multiple ACLs. A route map statement can contain multiple match commands. Logical or algorithm function is performed across all the match commands to reach a permit or deny decision.

For example:

```
match length A B
match ip address ac11 ac12
match ip address ac13
```

A packet is permitted if it is permitted by match length A B or ac11 or ac12 or ac13

- If the decision reached is permit, then the action specified by the set command is applied on the packet .

- If the decision reached is deny, then the PBR action (specified in the set command) is not applied. Instead the processing logic moves forward to look at the next route-map statement in the sequence (the statement with the next higher sequence number). If no next statement exists, PBR processing terminates, and the packet is routed using the default IP routing table.
- For PBR, route-map statements marked as deny are not supported.

You can use standard IP ACLs to specify match criteria for a source address or extended IP ACLs to specify match criteria based on an application, a protocol type, or an end station. The process proceeds through the route map until a match is found. If no match is found, normal destination-based routing occurs. There is an implicit deny at the end of the list of match statements.

If match clauses are satisfied, you can use a set clause to specify the IP addresses identifying the next hop router in the path.

How to Configure PBR

- To use PBR, you must have the feature set enabled on the switch or active stack.
- Multicast traffic is not policy-routed. PBR applies to only to unicast traffic.
- You can enable PBR on a routed port or an SVI.
- The switch supports PBR based on match length.
- You can apply a policy route map to an EtherChannel port channel in Layer 3 mode, but you cannot apply a policy route map to a physical interface that is a member of the EtherChannel. If you try to do so, the command is rejected. When a policy route map is applied to a physical interface, that interface cannot become a member of an EtherChannel.
- You can define a maximum of 128 IP policy route maps on the switch or switch stack.
- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch or switch stack.
- When configuring match criteria in a route map, follow these guidelines:
 - Do not match ACLs that permit packets destined for a local address. PBR would forward these packets, which could cause ping or Telnet failure or route protocol flapping.
- VRF and PBR are mutually exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. The reverse is also true, you cannot enable PBR when VRF is enabled on an interface.
- The number of hardware entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.
- PBR based on TOS, DSCP and IP Precedence are not supported.
- Set interface, set default next-hop and set default interface are not supported.
- **ip next-hop recursive** and **ip next-hop verify availability** features are not available and the next-hop should be directly connected.
- Policy-maps with no set actions are supported. Matching packets are routed normally.
- Policy-maps with no match clauses are supported. Set actions are applied to all packets.

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses are subject to PBR.

Packets that are generated by the switch, or local packets, are not normally policy-routed. When you globally enable local PBR on the switch, all packets that originate on the switch are subject to local PBR. Local PBR is disabled by default.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	route-map map-tag [permit] [sequence number] Example: Switch(config)# route-map pbr-map permit	Defines route maps that are used to control where packets are output, and enters route-map configuration mode. <ul style="list-style-type: none"> • <i>map-tag</i> — A meaningful name for the route map. The ip policy route-map interface configuration command uses this name to reference the route map. Multiple route-map statements with the same map tag define a single route map. • (Optional) permit — If permit is specified and the match criteria are met for this route map, the route is policy routed as defined by the set actions. • (Optional) <i>sequence number</i> — The sequence number shows the position of the route-map statement in the given route map.
Step 3	match ip address {access-list-number access-list-name} [access-list-number ...access-list-name] Example: Switch(config-route-map)# match ip address 110 140	Matches the source and destination IP addresses that are permitted by one or more standard or extended access lists. ACLs can match on more than one source and destination IP address. If you do not specify a match command, the route map is applicable to all packets.
Step 4	match length min max Example: Switch(config-route-map)# match length 64 1500	Matches the length of the packet.
Step 5	set ip next-hop ip-address [...ip-address] Example: Switch(config-route-map)# set ip next-hop 10.1.6.2	Specifies the action to be taken on the packets that match the criteria. Sets next hop to which to route the packet (the next hop must be adjacent).
Step 6	set ip next-hop verify-availability [next-hop-address sequence track object] Example:	Configures the route map to verify the reachability of the tracked object.

	Command or Action	Purpose
	Switch(config-route-map)# set ip next-hop verify-availability 95.1.1.2.1 track 100	Note This command is not supported on IPv6 and VRF.
Step 7	exit Example: Switch(config-route-map)# exit	Returns to global configuration mode.
Step 8	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to be configured.
Step 9	ip policy route-map <i>map-tag</i> Example: Switch(config-if)# ip policy route-map pbr-map	Enables PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in the order of sequence number until the first match. If there is no match, packets are routed as usual.
Step 10	ip route-cache policy Example: Switch(config-if)# ip route-cache policy	(Optional) Enables fast-switching PBR. You must enable PBR before enabling fast-switching PBR.
Step 11	exit Example: Switch(config-if)# exit	Returns to global configuration mode.
Step 12	ip local policy route-map <i>map-tag</i> Example: Switch(config)# ip local policy route-map local-pbr	(Optional) Enables local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch, and not to incoming packets.
Step 13	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 14	show route-map [<i>map-name</i>] Example: Switch# show route-map	(Optional) Displays all the route maps configured or only the one specified to verify configuration.
Step 15	show ip policy Example: Switch# show ip policy	(Optional) Displays policy route maps attached to the interface.
Step 16	show ip local policy Example: Switch# show ip local policy	(Optional) Displays whether or not local policy routing is enabled and, if so, the route map being used.

Filtering Routing Information

You can filter routing protocol information by performing the tasks described in this section.



Note When routes are redistributed between OSPF processes, no OSPF metrics are preserved.

Setting Passive Interfaces

To prevent other routers on a local network from dynamically learning about routes, you can use the **passive-interface** router configuration command to keep routing update messages from being sent through a router interface. When you use this command in the OSPF protocol, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface.

In networks with many interfaces, to avoid having to manually set them as passive, you can set all interfaces to be passive by default by using the **passive-interface default** router configuration command and manually setting interfaces where adjacencies are desired.

Use a network monitoring privileged EXEC command such as **show ip ospf interface** to verify the interfaces that you enabled as passive, or use the **show ip interface** privileged EXEC command to verify the interfaces that you enabled as active.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router { rip ospf eigrp } Example: Switch(config)# router ospf	Enters router configuration mode.
Step 3	passive-interface <i>interface-id</i> Example: Switch(config-router)# passive-interface gigabitethernet 1/0/1	Suppresses sending routing updates through the specified Layer 3 interface.
Step 4	passive-interface default Example: Switch(config-router)# passive-interface default	(Optional) Sets all interfaces as passive by default.
Step 5	no passive-interface <i>interface type</i> Example:	(Optional) Activates only those interfaces that need to have adjacencies sent.

	Command or Action	Purpose
	Switch(config-router)# no passive-interface gigabitethernet1/0/3 gigabitethernet 1/0/5	
Step 6	network <i>network-address</i> Example: Switch(config-router)# network 10.1.1.1	(Optional) Specifies the list of networks for the routing process. The <i>network-address</i> is an IP address.
Step 7	end Example: Switch(config-router)# end	Returns to privileged EXEC mode.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Controlling Advertising and Processing in Routing Updates

You can use the **distribute-list** router configuration command with access control lists to suppress routes from being advertised in routing updates and to prevent other routers from learning one or more routes. When used in OSPF, this feature applies to only external routes, and you cannot specify an interface name.

You can also use a **distribute-list** router configuration command to avoid processing certain routes listed in incoming updates. (This feature does not apply to OSPF.)

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router { rip eigrp } Example: Switch(config)# router eigrp 10	Enters router configuration mode.
Step 3	distribute-list {access-list-number access-list-name} out [<i>interface-name routing process autonomous-system-number</i>] Example: Switch(config-router)# distribute-list 120 out gigabitethernet 1/0/7	Permits or denies routes from being advertised in routing updates, depending upon the action listed in the access list.

	Command or Action	Purpose
Step 4	distribute-list <i>{access-list-number access-list-name}</i> in [<i>type-number</i>] Example: Switch(config-router)# distribute-list 125 in	Suppresses processing in routes listed in updates.
Step 5	end Example: Switch(config-router)# end	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Filtering Sources of Routing Information

Because some routing information might be more accurate than others, you can use filtering to prioritize information coming from different sources. An administrative distance is a rating of the trustworthiness of a routing information source, such as a router or group of routers. In a large network, some routing protocols can be more reliable than others. By specifying administrative distance values, you enable the router to intelligently discriminate between sources of routing information. The router always picks the route whose routing protocol has the lowest administrative distance.

Because each network has its own requirements, there are no general guidelines for assigning administrative distances.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	router { rip ospf eigrp } Example: Switch(config)# router eigrp 10	Enters router configuration mode.
Step 3	distance <i>weight</i> { <i>ip-address</i> { <i>ip-address mask</i> }} [<i>ip access list</i>] Example: Switch(config-router)# distance 50 10.1.5.1	Defines an administrative distance. <i>weight</i> —The administrative distance as an integer from 10 to 255. Used alone, <i>weight</i> specifies a default administrative distance that is used when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table.

	Command or Action	Purpose
		(Optional) <i>ip access list</i> —An IP standard or extended access list to be applied to incoming routing updates.
Step 4	end Example: <pre>Switch(config-router)# end</pre>	Returns to privileged EXEC mode.
Step 5	show ip protocols Example: <pre>Switch# show ip protocols</pre>	Displays the default administrative distance for a specified routing process.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Managing Authentication Keys

Key management is a method of controlling authentication keys used by routing protocols. Not all protocols can use key management. Authentication keys are available for EIGRP and RIP Version 2.

Prerequisites

Before you manage authentication keys, you must enable authentication. See the appropriate protocol section to see how to enable authentication for that protocol. To manage authentication keys, define a key chain, identify the keys that belong to the key chain, and specify how long each key is valid. Each key has its own key identifier (specified with the **key number** key chain configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and Message Digest 5 (MD5) authentication key in use.

How to Configure Authentication Keys

You can configure multiple keys with life times. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in order from lowest to highest, and uses the first valid key it encounters. The lifetimes allow for overlap during key changes. Note that the router must know these lifetimes.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	key chain <i>name-of-chain</i> Example: Switch(config)# key chain key10	Identifies a key chain, and enter key chain configuration mode.
Step 3	key number Example: Switch(config-keychain)# key 2000	Identifies the key number. The range is 0 to 2147483647.
Step 4	key-string <i>text</i> Example: Switch(config-keychain)# key-string Room 20, 10th floor	Identifies the key string. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters, but the first character cannot be a number.
Step 5	accept-lifetime <i>start-time</i> { infinite <i>end-time</i> duration <i>seconds</i> } Example: Switch(config-keychain)# accept-lifetime 12:30:00 Jan 25 1009 infinite	(Optional) Specifies the time period during which the key can be received. The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month date year</i> or <i>hh:mm:ss date Month year</i> . The default is forever with the default <i>start-time</i> and the earliest acceptable date as January 1, 1993. The default <i>end-time</i> and duration is infinite .
Step 6	send-lifetime <i>start-time</i> { infinite <i>end-time</i> duration <i>seconds</i> } Example: Switch(config-keychain)# accept-lifetime 23:30:00 Jan 25 1019 infinite	(Optional) Specifies the time period during which the key can be sent. The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month date year</i> or <i>hh:mm:ss date Month year</i> . The default is forever with the default <i>start-time</i> and the earliest acceptable date as January 1, 1993. The default <i>end-time</i> and duration is infinite .
Step 7	end Example: Switch(config-keychain)# end	Returns to privileged EXEC mode.
Step 8	show key chain Example: Switch# show key chain	Displays authentication key information.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining the IP Network

You can remove all contents of a particular cache, table, or database. You can also display specific statistics.

Table 112: Commands to Clear IP Routes or Display Route Status

Command	Purpose
show ip route [<i>address</i> [<i>mask</i>] [longer-prefixes]]	Displays the current state of the routing table.
show ip route summary	Displays the current state of the routing table in summary.
show platform ip unicast	Displays platform-dependent IP unicast information.



CHAPTER 48

Configuring Policy-Based Routing (PBR)

- [Policy-Based Routing, on page 1043](#)

Policy-Based Routing

Information About Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can specify and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

With PBR, you classify traffic using access control lists (ACLs) and then make traffic go through a different path. PBR is applied to incoming packets. All packets received on an interface with PBR enabled are passed through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop.

- Route map statement marked as permit is processed as follows:
 - A match command can match on length or multiple ACLs. A route map statement can contain multiple match commands. Logical or algorithm function is performed across all the match commands to reach a permit or deny decision.

For example:

```
match length A B
match ip address acl1 acl2
match ip address acl3
```

A packet is permitted if it is permitted by match length A B or acl1 or acl2 or acl3

- If the decision reached is permit, then the action specified by the set command is applied on the packet .
- If the decision reached is deny, then the PBR action (specified in the set command) is not applied. Instead the processing logic moves forward to look at the next route-map statement in the sequence (the statement with the next higher sequence number). If no next statement exists, PBR processing terminates, and the packet is routed using the default IP routing table.

- For PBR, route-map statements marked as deny are not supported.

You can use standard IP ACLs to specify match criteria for a source address or extended IP ACLs to specify match criteria based on an application, a protocol type, or an end station. The process proceeds through the route map until a match is found. If no match is found, normal destination-based routing occurs. There is an implicit deny at the end of the list of match statements.

If match clauses are satisfied, you can use a set clause to specify the IP addresses identifying the next hop router in the path.

How to Configure PBR

- To use PBR, you must have the feature set enabled on the switch or active stack.
- Multicast traffic is not policy-routed. PBR applies to only to unicast traffic.
- You can enable PBR on a routed port or an SVI.
- The switch supports PBR based on match length.
- You can apply a policy route map to an EtherChannel port channel in Layer 3 mode, but you cannot apply a policy route map to a physical interface that is a member of the EtherChannel. If you try to do so, the command is rejected. When a policy route map is applied to a physical interface, that interface cannot become a member of an EtherChannel.
- You can define a maximum of 128 IP policy route maps on the switch or switch stack.
- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch or switch stack.
- When configuring match criteria in a route map, follow these guidelines:
 - Do not match ACLs that permit packets destined for a local address. PBR would forward these packets, which could cause ping or Telnet failure or route protocol flapping.
- VRF and PBR are mutually exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. The reverse is also true, you cannot enable PBR when VRF is enabled on an interface.
- The number of hardware entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.
- PBR based on TOS, DSCP and IP Precedence are not supported.
- Set interface, set default next-hop and set default interface are not supported.
- **ip next-hop recursive** and **ip next-hop verify availability** features are not available and the next-hop should be directly connected.
- Policy-maps with no set actions are supported. Matching packets are routed normally.

- Policy-maps with no match clauses are supported. Set actions are applied to all packets.

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses are subject to PBR.

Packets that are generated by the switch, or local packets, are not normally policy-routed. When you globally enable local PBR on the switch, all packets that originate on the switch are subject to local PBR. Local PBR is disabled by default.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	route-map map-tag [permit] [sequence number] Example: Switch(config)# route-map pbr-map permit	Defines route maps that are used to control where packets are output, and enters route-map configuration mode. <ul style="list-style-type: none"> • <i>map-tag</i> – A meaningful name for the route map. The ip policy route-map interface configuration command uses this name to reference the route map. Multiple route-map statements with the same map tag define a single route map. • (Optional) permit – If permit is specified and the match criteria are met for this route map, the route is policy routed as defined by the set actions. • (Optional) <i>sequence number</i> – The sequence number shows the position of the route-map statement in the given route map.
Step 3	match ip address {access-list-number access-list-name} [access-list-number ...access-list-name] Example: Switch(config-route-map)# match ip address 110 140	Matches the source and destination IP addresses that are permitted by one or more standard or extended access lists. ACLs can match on more than one source and destination IP address. If you do not specify a match command, the route map is applicable to all packets.
Step 4	match length min max Example: Switch(config-route-map)# match length 64 1500	Matches the length of the packet.
Step 5	set ip next-hop ip-address [...ip-address] Example: Switch(config-route-map)# set ip next-hop 10.1.6.2	Specifies the action to be taken on the packets that match the criteria. Sets next hop to which to route the packet (the next hop must be adjacent).

	Command or Action	Purpose
Step 6	<p>set ip next-hop verify-availability [<i>next-hop-address sequence track object</i>]</p> <p>Example:</p> <pre>Switch(config-route-map)# set ip next-hop verify-availability 95.1.1.2.1 track 100</pre>	<p>Configures the route map to verify the reachability of the tracked object.</p> <p>Note This command is not supported on IPv6 and VRF.</p>
Step 7	<p>exit</p> <p>Example:</p> <pre>Switch(config-route-map)# exit</pre>	Returns to global configuration mode.
Step 8	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode, and specifies the interface to be configured.
Step 9	<p>ip policy route-map <i>map-tag</i></p> <p>Example:</p> <pre>Switch(config-if)# ip policy route-map pbr-map</pre>	Enables PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in the order of sequence number until the first match. If there is no match, packets are routed as usual.
Step 10	<p>ip route-cache policy</p> <p>Example:</p> <pre>Switch(config-if)# ip route-cache policy</pre>	(Optional) Enables fast-switching PBR. You must enable PBR before enabling fast-switching PBR.
Step 11	<p>exit</p> <p>Example:</p> <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 12	<p>ip local policy route-map <i>map-tag</i></p> <p>Example:</p> <pre>Switch(config)# ip local policy route-map local-pbr</pre>	(Optional) Enables local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch, and not to incoming packets.
Step 13	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 14	<p>show route-map [<i>map-name</i>]</p> <p>Example:</p> <pre>Switch# show route-map</pre>	(Optional) Displays all the route maps configured or only the one specified to verify configuration.
Step 15	<p>show ip policy</p> <p>Example:</p> <pre>Switch# show ip policy</pre>	(Optional) Displays policy route maps attached to the interface.

	Command or Action	Purpose
Step 16	show ip local policy Example: Switch# show ip local policy	(Optional) Displays whether or not local policy routing is enabled and, if so, the route map being used.

Feature Information for Configuring PBR

Table 113: Feature information for PBR

Feature Name	Releases	Feature Information
Policy-Based Routing	Cisco IOS Release 15.2(6)E2	Policy-based routing is used to configure a defined policy for traffic flows.



CHAPTER 49

Configuring EIGRP Stub Routing

- [EIGRP Stub Routing, on page 1049](#)

EIGRP Stub Routing

The EIGRP stub routing feature improves network stability, reduces resource utilization, and simplifies the stub device configuration.

Stub routing is commonly used in hub-and-spoke network topologies. In a hub-and-spoke network, one or more end (stub) networks are connected to a remote device (the spoke) that is connected to one or more distribution devices (the hub). The remote device is adjacent to one or more distribution devices. The only route for IP traffic to reach the remote device is through a distribution device.

Information About EIGRP Stub Routing

EIGRP Stub Routing

The EIGRP stub routing feature improves network stability, reduces resource utilization, and simplifies the stub device configuration.

Stub routing is commonly used in hub-and-spoke network topologies. In a hub-and-spoke network, one or more end (stub) networks are connected to a remote device (the spoke) that is connected to one or more distribution devices (the hub). The remote device is adjacent to one or more distribution devices. The only route for IP traffic to reach the remote device is through a distribution device. This type of configuration is commonly used in WAN topologies, where the distribution device is directly connected to a WAN. The distribution device can be connected to many remote devices, which is often the case. In a hub-and-spoke topology, the remote device must forward all nonlocal traffic to a distribution device, so it becomes unnecessary for the remote device to have a complete routing table. Generally, the distribution device need not send anything more than a default route to the remote device.

When using the EIGRP stub routing feature, you need to configure the distribution and remote devices to use EIGRP and configure only the remote device as a stub. Only specified routes are propagated from the remote (stub) device. The stub device responds to all queries for summaries, connected routes, redistributed static routes, external routes, and internal routes with the message “inaccessible.” A device that is configured as a stub will send a special peer information packet to all neighboring devices to report its status as a stub device.

Any neighbor that receives a packet informing it of the stub status will not query the stub device for any routes, and a device that has a stub peer will not query that peer. The stub device will depend on the distribution device to send proper updates to all peers.

The figure below shows a simple hub-and-spoke network.

The stub routing feature by itself does not prevent routes from being advertised to the remote device. In the above example, the remote device can access the corporate network and the Internet only through the distribution device. Having a complete route table on the remote device would serve no functional purpose because the path to the corporate network and the Internet would always be through the distribution device. The large route table would only reduce the amount of memory required by the remote device. Bandwidth and memory can be conserved by summarizing and filtering routes in the distribution device. The remote device need not receive routes that have been learned from other networks because the remote device must send all nonlocal traffic, regardless of the destination, to the distribution device. If a true stub network is desired, the distribution device should be configured to send only a default route to the remote device. The EIGRP stub routing feature does not automatically enable summarization on distribution devices. In most cases, the network administrator will need to configure summarization on distribution devices.



Note When configuring the distribution device to send only a default route to the remote device, you must use the **ip classless** command on the remote device. By default, the **ip classless** command is enabled in all Cisco images that support the EIGRP stub routing feature.

Without the EIGRP stub routing feature, even after routes that are sent from the distribution device to the remote device have been filtered or summarized, a problem might occur. If a route is lost somewhere in the corporate network, EIGRP could send a query to the distribution device, which in turn would send a query to the remote device, even if routes are being summarized. If there is a communication problem (over the WAN link) between the distribution device and the remote device, an EIGRP stuck in active (SIA) condition could occur and cause instability elsewhere in the network. The EIGRP stub routing feature allows a network administrator to prevent queries from being sent to the remote device.

Dual-Homed Remote Topology

In addition to a simple hub-and-spoke network, where a remote device is connected to a single distribution device, the remote device can be dual-homed to two or more distribution devices. This configuration adds redundancy and introduces unique issues, and the stub feature helps to address some of these issues.

A dual-homed remote device will have two or more distribution (hub) devices. However, the principles of stub routing are the same as they are with a hub-and-spoke topology. The figure below shows a common dual-homed remote topology with one remote device: however, 100 or more devices could be connected on the same interfaces on distribution Device 1 and distribution Device 2. The remote device will use the best route to reach its destination. If distribution Device 1 experiences a failure, the remote device can still use distribution Device 2 to reach the corporate network.

The figure above shows a simple dual-homed remote topology with one remote device and two distribution devices. Both distribution devices maintain routes to the corporate network and stub network 10.1.1.0/24.

Dual-homed routing can introduce instability into an EIGRP network. In the figure below, distribution Device 1 is directly connected to network 10.3.1.0/24. If summarization or filtering is applied on distribution Device 1, the device will advertise network 10.3.1.0/24 to all of its directly connected EIGRP neighbors (distribution Device 2 and the remote device).

The figure above shows a simple dual-homed remote topology, where distribution Device 1 is connected to both network 10.3.1.0/24 and network 10.2.1.0/24.

If the 10.2.1.0/24 link between distribution Device 1 and distribution Device 2 fails, the lowest cost path to network 10.3.1.0/24 from distribution Device 2 will be through the remote device (see the figure below). This route is not desirable because the traffic that was previously traveling across the corporate network 10.2.1.0/24 would now be sent across a much lower bandwidth connection. The overutilization of the lower bandwidth WAN connection can cause many problems that might affect the entire corporate network. The use of the lower bandwidth route that passes through the remote device may cause WAN EIGRP distribution devices to be dropped. Serial lines on distribution and remote devices may also be dropped, and EIGRP SIA errors on the distribution and core devices can occur.

It is not desirable for traffic from distribution Device 2 to travel through any remote device to reach network 10.3.1.0/24. Backup routes can be used if links are sized to manage the load. However, most networks, of the type shown in the figure above, have remote devices located at remote offices with relatively slow links. To ensure that traffic from distribution devices are not routed through a remote device, you can configure route summarization on the distribution device and the remote device.

It is typically undesirable for traffic from a distribution device to use a remote device as a transit path. A typical connection from a distribution device to a remote device would have much less bandwidth than a connection at the network core. Attempting to use a remote device with a limited bandwidth connection as a transit path would generally produce excessive congestion at the remote device. The EIGRP stub routing feature can prevent this problem by preventing the remote device from advertising core routes back to the distribution devices. In the above example, routes learned by the remote device from distribution Device 1 will not be advertised to distribution Device 2. Therefore, distribution Device 2 will not use the remote device as a transit for traffic destined to the network core.

The EIGRP stub routing feature provides network stability. If the network is not stable, this feature prevents EIGRP queries from being sent over limited bandwidth links to nontransit devices. Instead, distribution devices to which the stub device is connected answer queries on behalf of the stub device. This feature greatly reduces the chance of further network instability due to congested or problematic WAN links. The EIGRP stub routing feature also simplifies the configuration and maintenance of hub-and-spoke networks. When stub routing is enabled in dual-homed remote configurations, it is no longer necessary to configure filtering on remote devices to prevent those devices from appearing as transit paths to hub devices.

**Caution**

The EIGRP stub routing feature should be used only on stub devices. A stub device is defined as a device connected to the network core or distribution layer through which core transit traffic should not flow. A stub device should not have any EIGRP neighbors other than distribution devices. Ignoring this restriction will cause undesirable behavior.

**Note**

Multiaccess interfaces such as ATM, Gigabit Ethernet, Frame Relay, ISDN PRI, and X.25 are supported by the EIGRP stub routing feature only when all devices on that interface, except the hub, are configured as stub devices.

How to Configure EIGRP Stub Routing

Configuring the EIGRP Stub Routing Autonomous System Configuration

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *autonomous-system-number*
4. **network** *ip-address* [**wildcard-mask**]
5. **eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static**] [**summary**] [**redistributed**]
6. **end**
7. **show ip eigrp neighbors** [*interface-type* | *as-number* | **static** | **detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router eigrp <i>autonomous-system-number</i> Example: Device(config)# router eigrp 1	Configures a remote or distribution device to run an EIGRP process and enters router configuration mode.
Step 4	network <i>ip-address</i> [wildcard-mask] Example: Device(config-router)# network 172.16.0.0	Specifies the network address of the EIGRP distribution device.
Step 5	eigrp stub [receive-only] [leak-map <i>name</i>] [connected] [static] [summary] [redistributed] Example: Device(config-router)# eigrp stub connected static	Configures a remote device as an EIGRP stub device.
Step 6	end Example: Device(config-router)# end	Exits router configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show ip eigrp neighbors [<i>interface-type</i> <i>as-number</i> static detail] Example: Device# show ip eigrp neighbors detail	(Optional) Verifies that a remote device has been configured as a stub device with EIGRP. <ul style="list-style-type: none"> • Enter this command on the distribution device. The last line of the output displays the stub status of the remote or spoke device.

Configuring the EIGRP Stub Routing Named Configuration

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *virtual-instance-name*
4. Enter one of the following:
 - **address-family ipv4** [**multicast**] [**unicast**] [**vrf vrf-name**] **autonomous-system** *autonomous-system-number*
 - **address-family ipv6** [**unicast**] [**vrf vrf-name**] **autonomous-system** *autonomous-system-number*
5. **network** *ip-address* [**wildcard-mask**]
6. **eigrp stub** [**receive-only**] [**leak-map name**] [**connected**] [**static**] [**summary**] [**redistributed**]
7. **exit-address-family**
8. **end**
9. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf vrf-name**] [*autonomous-system-number*] [**multicast**] [**neighbors**] [**static**] [**detail**] [*interface-type interface-number*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router eigrp <i>virtual-instance-name</i> Example: Device(config)# router eigrp virtual-name1	Enables an EIGRP routing process and enters router configuration mode.
Step 4	Enter one of the following: <ul style="list-style-type: none"> • address-family ipv4 [multicast] [unicast] [vrf vrf-name] autonomous-system <i>autonomous-system-number</i> 	Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance.

	Command or Action	Purpose
	<ul style="list-style-type: none"> • address-family ipv6 [unicast] [vrf vrf-name] autonomous-system autonomous-system-number <p>Example:</p> <pre>Device(config-router)# address-family ipv4 autonomous-system 45000 Device(config-router)# address-family ipv6 autonomous-system 45000</pre>	
Step 5	<p>network ip-address [wildcard-mask]</p> <p>Example:</p> <pre>Device(config-router-af)# network 172.16.0.0</pre>	Specifies the network address of the EIGRP distribution device.
Step 6	<p>eigrp stub [receive-only] [leak-map name] [connected] [static] [summary] [redistributed]</p> <p>Example:</p> <pre>Device(config-router-af) eigrp stub leak-map map1</pre>	Configures a device as a stub using EIGRP.
Step 7	<p>exit-address-family</p> <p>Example:</p> <pre>Device(config-router-af)# exit-address-family</pre>	Exits address family configuration mode.
Step 8	<p>end</p> <p>Example:</p> <pre>Device(config-router)# end</pre>	Exits router configuration mode and returns to privileged EXEC mode.
Step 9	<p>show eigrp address-family {ipv4 ipv6} [vrf vrf-name] [autonomous-system-number] [multicast] [neighbors] [static] [detail] [interface-type interface-number]</p> <p>Example:</p> <pre>Device# show eigrp address-family ipv4 neighbors detail</pre>	(Optional) Displays neighbors discovered by EIGRP.

Configuration Examples for EIGRP Stub Routing

Example: EIGRP Stub Routing—Autonomous System Configuration

A device that is configured as a stub with the **eigrp stub** command shares connected and summary routing information with all neighbor devices by default. The following six keywords can be used with the **eigrp stub** command to modify this behavior:

- **connected**
- **leak-map**
- **receive-only**
- **redistributed**

- **static**
- **summary**

This section provides configuration examples for all forms of the **eigrp stub** command for an EIGRP autonomous system configuration.

Example: eigrp stub Command

In the following example, the **eigrp stub** command is used to configure the device as a stub that advertises connected and summary routes:

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub
```

Example: eigrp stub connected static Command

In the following example, the **eigrp stub** command is used with the **connected** and **static** keywords to configure the device as a stub that advertises connected and static routes (sending summary routes will not be permitted):

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub connected static
```

Example: eigrp stub leak-map Command

In the following example, the **eigrp stub** command is issued with the **leak-map** *name* keyword-argument pair to configure the device to reference a leak map that identifies routes that would have been suppressed:

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub leak-map map1
```

Example: eigrp stub receive-only Command

In the following example, the **eigrp stub** command is issued with the **receive-only** keyword to configure the device as a receive-only neighbor (connected, summary, and static routes will not be sent):

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub receive-only
```

Example: eigrp stub redistributed Command

In the following example, the **eigrp stub** command is issued with the **redistributed** keyword to configure the device to advertise other protocols and autonomous systems:

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub redistributed
```

Example: EIGRP Stub Routing—Named Configuration

A device that is configured as a stub with the **eigrp stub** command shares connected and summary routing information with all neighbor devices by default. The following six keywords can be used with the **eigrp stub** command to modify this behavior:

- **connected**
- **leak-map**
- **receive-only**
- **redistributed**
- **static**
- **summary**

This section provides configuration examples for all forms of the **eigrp stub** command for an EIGRP named configuration.

Example: eigrp stub Command

In the following example, the **eigrp stub** command is used to configure the device as a stub that advertises connected and summary routes:

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af) eigrp stub
```

Example: eigrp stub connected static Command

In the following named configuration example, the **eigrp stub** command is issued with the **connected** and **static** keywords to configure the device as a stub that advertises connected and static routes (sending summary routes will not be permitted):

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# eigrp stub connected static
```

Example: eigrp stub leak-map Command

In the following named configuration example, the **eigrp stub** command is issued with the **leak-map name** keyword-argument pair to configure the device to reference a leak map that identifies routes that would normally have been suppressed:


```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# eigrp stub leak-map map1
```

Example: eigrp stub receive-only Command

In the following named configuration example, the **eigrp stub** command is issued with the **receive-only** keyword to configure the device as a receive-only neighbor (connected, summary, and static routes will not be sent):

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# eigrp stub receive-only
```

Example: eigrp stub redistributed Command

In the following named configuration example, the **eigrp stub** command is issued with the **redistributed** keyword to configure the device to advertise other protocols and autonomous systems:

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# eigrp stub redistributed
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
EIGRP commands	Cisco IOS IP Routing: EIGRP Command Reference
EIGRP FAQ	EIGRP Frequently Asked Questions
EIGRP Technology White Papers	Enhanced Interior Gateway Routing Protocol

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for EIGRP Stub Routing

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 114: Feature Information for EIGRP Stub Routing

Feature Name	Releases	Feature Information
EIGRP Stub Routing	Cisco IOS XE 15.2(6)E2	The EIGRP Stub Routing feature improves network stability, reduces resource utilization, and simplifies stub router configuration. Stub routing is commonly used in a hub-and-spoke network topology. In a hub-and-spoke network, one or more end (stub) networks are connected to a remote router (the spoke) that is connected to one or more distribution routers (the hub). The remote router is adjacent only to one or more distribution routers.



PART IX

Security

- [Security Features Overview](#), on page 1061
- [Preventing Unauthorized Access](#) , on page 1065
- [Controlling Switch Access with Passwords and Privilege Levels](#) , on page 1067
- [Configuring TACACS+](#) , on page 1085
- [Configuring RADIUS](#) , on page 1099
- [Configuring Kerberos](#) , on page 1141
- [Configuring Local Authentication and Authorization](#) , on page 1147
- [Configuring Secure Shell](#) , on page 1151
- [Configuring SSH File Transfer Protocol](#), on page 1159
- [X.509v3 Certificates for SSH Authentication](#), on page 1163
- [Configuring Secure Socket Layer HTTP](#) , on page 1175
- [Certification Authority Interoperability](#), on page 1187
- [Access Control List Overview](#), on page 1205
- [IPv4 ACLs](#) , on page 1215
- [IPv6 ACLs](#), on page 1267
- [Configuring DHCP](#) , on page 1277
- [Configuring IP Source Guard](#) , on page 1299
- [Configuring Dynamic ARP Inspection](#), on page 1307
- [Configuring IEEE 802.1x Port-Based Authentication](#), on page 1323
- [Configuring MACsec Encryption](#), on page 1413
- [Web-Based Authentication](#) , on page 1451
- [Auto Identity](#), on page 1477
- [Configuring Port-Based Traffic Control](#), on page 1489
- [Configuring IPv6 First Hop Security](#), on page 1517

- [Configuring FIPS, on page 1551](#)
- [Configuring Control Plane Policing, on page 1553](#)



CHAPTER 50

Security Features Overview

- [Security Features Overview](#), on page 1061

Security Features Overview

The security features are as follows:

- IPv6 First Hop Security—A suite of security features to be applied at the first hop switch to protect against vulnerabilities inherent in IPv6 networks. These include, Binding Integrity Guard (Binding Table), Router Advertisement Guard (RA Guard), DHCP Guard, IPv6 Neighbor Discovery Inspection (ND Guard).
- Web Authentication—Allows a supplicant (client) that does not support IEEE 802.1x functionality to be authenticated using a web browser.
- Local Web Authentication Banner—A custom banner or an image file displayed at a web authentication login screen.
- IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute
- Password-protected access (read-only and read-write access) to management interfaces (device manager, Network Assistant, and the CLI) for protection against unauthorized configuration changes
- Multilevel security for a choice of security level, notification, and resulting actions
- Static MAC addressing for ensuring security
- Protected port option for restricting the forwarding of traffic to designated ports on the same switch
- Port security option for limiting and identifying MAC addresses of the stations allowed to access the port
- VLAN aware port security option to shut down the VLAN on the port when a violation occurs, instead of shutting down the entire port.
- Port security aging to set the aging time for secure addresses on a port.
- Protocol storm protection to control the rate of incoming protocol traffic to a switch by dropping packets that exceed a specified ingress rate.
- BPDU guard for shutting down a Port Fast-configured port when an invalid configuration occurs.

- Standard and extended IP access control lists (ACLs) for defining inbound security policies on Layer 2 interfaces (port ACLs).
- Extended MAC access control lists for defining security policies in the inbound direction on Layer 2 interfaces.
- Source and destination MAC-based ACLs for filtering non-IP traffic.
- DHCP snooping to filter untrusted DHCP messages between untrusted hosts and DHCP servers.
- IP source guard to restrict traffic on nonrouted interfaces by filtering traffic based on the DHCP snooping database and IP source bindings.
- Dynamic ARP inspection to prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN.
- IEEE 802.1x port-based authentication to prevent unauthorized devices (clients) from gaining access to the network. These 802.1x features are supported:
 - Support for single-host, multi-host, multi-auth, and multi-domain-auth modes.
 - Multidomain authentication (MDA) to allow both a data device and a voice device, such as an IP phone (Cisco or non-Cisco), to independently authenticate on the same IEEE 802.1x-enabled switch port.
 - Dynamic voice virtual LAN (VLAN) for MDA to allow a dynamic voice VLAN on an MDA-enabled port.
 - VLAN assignment for restricting 802.1x-authenticated users to a specified VLAN.
 - Support for VLAN assignment on a port configured for multi-auth mode. The RADIUS server assigns a VLAN to the first host to authenticate on the port, and subsequent hosts use the same VLAN. Voice VLAN assignment is supported for one IP phone.
 - Port security for controlling access to 802.1x ports.
 - Voice VLAN to permit a Cisco IP Phone to access the voice VLAN regardless of the authorized or unauthorized state of the port.
 - IP phone detection enhancement to detect and recognize a Cisco IP phone.
 - Guest VLAN to provide limited services to non-802.1x-compliant users.
 - Restricted VLAN to provide limited services to users who are 802.1x compliant, but do not have the credentials to authenticate via the standard 802.1x processes.
 - 802.1x accounting to track network usage.
 - 802.1x with wake-on-LAN to allow dormant PCs to be powered on based on the receipt of a specific Ethernet frame.
 - 802.1x readiness check to determine the readiness of connected end hosts before configuring IEEE 802.1x on the switch.
 - Voice aware 802.1x security to apply traffic violation actions only on the VLAN on which a security violation occurs.
 - MAC authentication bypass (MAB) to authorize clients based on the client MAC address.

- Network Admission Control (NAC) Layer 2 802.1x validation of the antivirus condition or posture of endpoint systems or clients before granting the devices network access.



Note NAC is not supported on LanLite images.

- Network Edge Access Topology (NEAT) with 802.1X switch supplicant, host authorization with CISP, and auto enablement to authenticate a switch outside a wiring closet as a supplicant to another switch.



Note NEAT is not supported on LanLite images.

- IEEE 802.1x with open access to allow a host to access the network before being authenticated.



Note This feature is not supported on LanLite images.

- IEEE 802.1x authentication with downloadable ACLs and redirect URLs to allow per-user ACL downloads from a Cisco Secure ACS server to an authenticated switch.
- Support for dynamic creation or attachment of an auth-default ACL on a port that has no configured static ACLs.



Note This feature is not supported on LanLite images.

- Flexible-authentication sequencing to configure the order of the authentication methods that a port tries when authenticating a new host.
- Multiple-user authentication to allow more than one host to authenticate on an 802.1x-enabled port.
- TACACS+, a proprietary feature for managing network security through a TACACS server for both IPv4 and IPv6.
- RADIUS for verifying the identity of, granting access to, and tracking the actions of remote users through authentication, authorization, and accounting (AAA) services for both IPv4 and IPv6.
- Enhancements to RADIUS, TACACS+, and SSH to function over IPv6.
- Secure Socket Layer (SSL) Version 3.0 support for the HTTP 1.1 server authentication, encryption, and message integrity and HTTP client authentication to allow secure HTTP communications (requires the cryptographic version of the software).
- IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute.
- Support for IP source guard on static hosts.
- RADIUS Change of Authorization (CoA) to change the attributes of a certain session after it is authenticated. When there is a change in policy for a user or user group in AAA, administrators can send

the RADIUS CoA packets from the AAA server, such as Cisco Identity Services Engine, or Cisco Secure ACS to reinitialize authentication, and apply to the new policies.

- IEEE 802.1x User Distribution to allow deployments with multiple VLANs (for a group of users) to improve scalability of the network by load balancing users across different VLANs. Authorized users are assigned to the least populated VLAN in the group, assigned by RADIUS server.



Note This feature is not supported on LanLite images.

- Support for critical VLAN—multi-host/multi-auth enabled ports are placed in a critical VLAN in order to permit access to critical resources if AAA server becomes unreachable.



Note This feature is not supported on LanLite images.

- Support for Network Edge Access Topology (NEAT) to change the port host mode and to apply a standard port configuration on the authenticator switch port.
- VLAN-ID based MAC authentication to use the combined VLAN and MAC address information for user authentication to prevent network access from unauthorized VLANs.
- MAC move to allow hosts (including the hosts connected behind an IP phone) to move across ports within the same switch without any restrictions to enable mobility. With MAC move, the switch treats the reappearance of the same MAC address on another port in the same way as a completely new MAC address.
- Support for 3DES and AES with version 3 of the Simple Network Management Protocol (SNMPv3). This release adds support for the 168-bit Triple Data Encryption Standard (3DES) and the 128-bit, 192-bit, and 256-bit Advanced Encryption Standard (AES) encryption algorithms to SNMPv3.
- Support for Cisco TrustSec SXP protocol. This feature is not supported on LanLite images.



CHAPTER 51

Preventing Unauthorized Access

- [Preventing Unauthorized Access, on page 1065](#)

Preventing Unauthorized Access

You can prevent unauthorized users from reconfiguring your switch and viewing configuration information. Typically, you want network administrators to have access to your switch while you restrict access to users who dial from outside the network through an asynchronous port, connect from outside the network through a serial port, or connect through a terminal or workstation from within the local network.

To prevent unauthorized access into your switch, you should configure one or more of these security features:

- At a minimum, you should configure passwords and privileges at each switch port. These passwords are locally stored on the switch. When users attempt to access the switch through a port or line, they must enter the password specified for the port or line before they can access the switch.
- For an additional layer of security, you can also configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.
- If you want to use username and password pairs, but you want to store them centrally on a server instead of locally, you can store them in a database on a security server. Multiple networking devices can then use the same database to obtain user authentication (and, if necessary, authorization) information.
- You can also enable the login enhancements feature, which logs both failed and unsuccessful login attempts. Login enhancements can also be configured to block future login attempts after a set number of unsuccessful attempts are made. For more information, see the Cisco IOS Login Enhancements documentation.



CHAPTER 52

Controlling Switch Access with Passwords and Privilege Levels

- [Restrictions for Controlling Switch Access with Passwords and Privileges, on page 1067](#)
- [Information About Passwords and Privilege Levels, on page 1068](#)
- [How to Control Switch Access with Passwords and Privilege Levels, on page 1070](#)
- [Monitoring Switch Access, on page 1082](#)
- [Configuration Examples for Setting Passwords and Privilege Levels, on page 1082](#)

Restrictions for Controlling Switch Access with Passwords and Privileges

The following are the restrictions for controlling switch access with passwords and privileges:

- Disabling password recovery will not work if you have set the switch to boot up manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:~*) after the switch is power cycled.

Restrictions and Guidelines for Reversible Password Types

If the startup configuration has a type 6 password and you downgrade to a version in which type 6 password is not supported, you can/may be locked out of the device.

Restrictions and Guidelines for Irreversible Password Types

- Username secret password type 5 and enable secret password type 5 must be migrated to the stronger password type 8 or 9. For more information, see [Protecting Enable and Enable Secret Passwords with Encryption, on page 1071](#).
- Plain text passwords are converted to nonreversible encrypted password type 9.



Note This is supported in Cisco IOS Release 15.2(7)E3 and later releases.

Information About Passwords and Privilege Levels

Default Password and Privilege Level Configuration

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

This table shows the default password and privilege level configuration.

Table 115: Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Additional Password Security

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

Masked Secret Password



Note This feature is supported on Cisco Catalyst 3560-CX Series Switches only.

With **enable secret** command, password is encrypted but is visible on the terminal when you type the password. To mask the password on the terminal, use the **masked-secret** global configuration command. The encryption type for this password is type 9, by default.

You can use this command to configure masked secret password for common criteria policy.

Password Recovery

By default, any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

To re-enable password recovery, use the **service password-recovery** global configuration command.

Terminal Line Telnet Configuration

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it when you set a Telnet password for a terminal line.

Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Privilege Levels

Cisco devices use privilege levels to provide password security for different levels of switch operation. By default, the Cisco IOS software operates in two modes (privilege levels) of password security: user EXEC (Level 1) and privileged EXEC (Level 15). You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

Privilege Levels on Lines

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

Command Privilege Levels

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

How to Control Switch Access with Passwords and Privilege Levels

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Follow these steps to set or change a static enable password:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **enable password** *password*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	enable password <i>password</i> Example:	Defines a new password or changes an existing password for access to privileged EXEC mode. By default, no password is defined.

	Command or Action	Purpose
	Switch(config)# enable password secret321	For <i>password</i> , specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Ctrl-v when you create the password; for example, to create the password abc?123, do this: <ol style="list-style-type: none"> a. Enter abc. b. Enter Ctrl-v. c. Enter ?123. When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Protecting Enable and Enable Secret Passwords with Encryption

Follow these steps to establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Use one of the following:
 - `enable password [level level] {password encryption-type encrypted-password}`
 - `enable secret [level level]`

{password encryption-type encrypted-password}

4. **service password-encryption**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>Use one of the following:</p> <ul style="list-style-type: none"> • <code>enable password [level level]</code> <i>{password encryption-type encrypted-password}</i> • <code>enable secret [level level]</code> <i>{password encryption-type encrypted-password}</i> <p>Example:</p> <pre>Switch(config)# enable password example102</pre> <p>or</p> <pre>Switch(config)# enable secret level 1 password secret123sample</pre>	<ul style="list-style-type: none"> • Defines a new password or changes an existing password for access to privileged EXEC mode. • Defines a secret password, which is saved using a nonreversible encryption method. <ul style="list-style-type: none"> • (Optional) For <i>level</i>, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges). • For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined. • (Optional) For <i>encryption-type</i>, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration. <p>Note If you specify an encryption type and then enter a clear text password, you can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.</p>

	Command or Action	Purpose
Step 4	service password-encryption Example: <pre>Switch(config)# service password-encryption</pre>	(Optional) Encrypts the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Masked Secret Password

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Use one of the following:
 - **username *name*masked-secret**
 - **username *name*common-criteria-policy *policy-name* masked-secret**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password, if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • username <i>name</i> masked-secret • username <i>name</i> common-criteria-policy <i>policy-name</i> masked-secret Example: Device(config) # username cisco masked-secret or Device(config) # username common-criteria-policy test-policy masked-secret	<ul style="list-style-type: none"> • Defines a masked secret password, which is saved using a nonreversible encryption method. • Defines a masked secret password for common criteria policy. <ul style="list-style-type: none"> • The masked secret password must be greater than 4 characters. The maximum length of masked-secret password is 256 characters. By default, no password is defined.
Step 4	end Example: Device(config) # end	Exits global configuration mode and returns to privileged EXEC mode.

Disabling Password Recovery

Follow these steps to disable password recovery to protect the security of your switch:

Before you begin

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **system disable password recovery switch {all | <1-9>}**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> <code>enable</code>	
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	system disable password recovery switch {all <1-9>} Example: Switch(config)# <code>system disable password recovery switch all</code>	Disables password recovery. <ul style="list-style-type: none"> • <i>all</i> - Sets the configuration on switches in stack. • <i><1-9></i> - Sets the configuration on the Switch Number selected. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.

What to do next

To remove **disable password recovery**, use the **no system disable password recovery switch all** global configuration command.

Setting a Telnet Password for a Terminal Line

Beginning in user EXEC mode, follow these steps to set a Telnet password for the connected terminal line:

Before you begin

- Attach a PC or workstation with emulation software to the switch console port, or attach a PC to the Ethernet management port.
- The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `line vty 0 15`
4. `password password`

5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Note If a password is required for access to privileged EXEC mode, you will be prompted for it. Enters privileged EXEC mode.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	line vty 0 15 Example: <pre>Switch(config)# line vty 0 15</pre>	Configures the number of Telnet sessions (lines), and enters line configuration mode. There are 16 possible sessions on a command-capable Switch. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 4	password <i>password</i> Example: <pre>Switch(config-line)# password abcxyz543</pre>	Sets a Telnet password for the line or lines. For <i>password</i> , specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	end Example: <pre>Switch(config-line)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Username and Password Pairs

Follow these steps to configure username and password pairs:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **username name [privilege level] {password encryption-type password}**
4. Use one of the following:
 - **line console 0**
 - **line vty 0 15**
5. **login local**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	username name [privilege level] {password encryption-type password} Example: <pre>Switch(config)# username adamsample privilege 1 password secret456</pre> <pre>Switch(config)# username 111111111111 mac attribute</pre>	Sets the username, privilege level, and password for each user. <ul style="list-style-type: none"> • For <i>name</i>, specify the user ID as one word or the MAC address. Spaces and quotation marks are not allowed. • You can configure a maximum of 12000 clients each, for both username and MAC filter. • (Optional) For <i>level</i>, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For <i>encryption-type</i>, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>password</i>, specify the password the user must enter to gain access to the Switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.
Step 4	Use one of the following: <ul style="list-style-type: none"> line console 0 line vty 0 15 Example: <pre>Switch(config)# line console 0</pre> or <pre>Switch(config)# line vty 15</pre>	Enters line configuration mode, and configures the console port (line 0) or the VTY lines (line 0 to 15).
Step 5	login local Example: <pre>Switch(config-line)# login local</pre>	Enables local password checking at login time. Authentication is based on the username specified in Step 3.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting the Privilege Level for a Command

Follow these steps to set the privilege level for a command:

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **privilege mode level level command**
4. **enable password level level password**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	privilege mode level level command Example: <pre>Switch(config)# privilege exec level 14 configure</pre>	Sets the privilege level for a command. <ul style="list-style-type: none"> • For <i>mode</i>, enter configure for global configuration mode, exec for EXEC mode, interface for interface configuration mode, or line for line configuration mode. • For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password. • For <i>command</i>, specify the command to which you want to restrict access.
Step 4	enable password level level password Example: <pre>Switch(config)# enable password level 14 SecretPswd14</pre>	Specifies the password to enable the privilege level. <ul style="list-style-type: none"> • For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Changing the Default Privilege Level for Lines

Follow these steps to change the default privilege level for the specified line:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty *line***
4. **privilege level *level***
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	line vty <i>line</i> Example: <pre>Switch(config)# line vty 10</pre>	Selects the virtual terminal line on which to restrict access.
Step 4	privilege level <i>level</i> Example: <pre>Switch(config)# privilege level 15</pre>	Changes the default privilege level for the line. For <i>level</i> , the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password.

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

Logging into and Exiting a Privilege Level

Beginning in user EXEC mode, follow these steps to log into a specified privilege level and exit a specified privilege level.

SUMMARY STEPS

1. **enable** *level*
2. **disable** *level*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable <i>level</i> Example: <pre>Switch> enable 15</pre>	Logs in to a specified privilege level. Following the example, Level 15 is privileged EXEC mode. For <i>level</i> , the range is 0 to 15.
Step 2	disable <i>level</i> Example: <pre>Switch# disable 1</pre>	Exits to a specified privilege level. Following the example, Level 1 is user EXEC mode. For <i>level</i> , the range is 0 to 15.

Monitoring Switch Access

Table 116: Commands for Displaying DHCP Information

<code>show privilege</code>	Displays the privilege level configuration.
-----------------------------	---------------------------------------------

Configuration Examples for Setting Passwords and Privilege Levels

Example: Setting or Changing a Static Enable Password

This example shows how to change the enable password to *11u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Switch(config)# enable password 11u2c3k4y5
```

Example: Protecting Enable and Enable Secret Passwords with Encryption

This example shows how to configure the encrypted password *\$1\$FaD0\$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Example: Configuring Masked Secret Password

The following example shows how to configure the masked secret password:

```
Device> enable
Device# configure terminal
Device(config)# username cisco masked-secret
Enter secret: *****
Confirm secret: *****
```

The following example shows how to configure the masked secret password for common criteria policy:

```
Device> enable
Device# configure terminal
Device(config)# username cisco common-criteria-policy test-policy masked-secret
Enter secret: *****
Confirm secret: *****
```

Example: Setting a Telnet Password for a Terminal Line

This example shows how to set the Telnet password to *let45me67in89*:

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

Example: Setting the Privilege Level for a Command

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Switch(config)# privilege exec level 14 configure
Switch(config)# enable password level 14 SecretPswd14
```




CHAPTER 53

Configuring TACACS+

- [Finding Feature Information](#), on page 1085
- [Prerequisites for TACACS+](#), on page 1085
- [Information About TACACS+](#), on page 1086
- [How to Configure Switch Access with TACACS+](#), on page 1090
- [Monitoring TACACS+](#), on page 1097

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for TACACS+

The following are the prerequisites for set up and configuration of switch access with TACACS+ (must be performed in the order presented):

1. Configure the switches with the TACACS+ server addresses.
2. Set an authentication key.
3. Configure the key from Step 2 on the TACACS+ servers.
4. Enable authentication, authorization, and accounting (AAA).
5. Create a login authentication method list.
6. Apply the list to the terminal lines.
7. Create an authorization and accounting method list.

The following are the prerequisites for controlling switch access with TACACS+:

- You must have access to a configured TACACS+ server to configure TACACS+ features on your switch. Also, you must have access to TACACS+ services maintained in a database on a TACACS+ daemon typically running on a LINUX or Windows workstation.
- We recommend a redundant connection between a switch stack and the TACACS+ server. This is to help ensure that the TACACS+ server remains accessible in case one of the connected stack members is removed from the switch stack.
- You need a system running the TACACS+ daemon software to use TACACS+ on your switch.
- To use TACACS+, it must be enabled.
- Authorization must be enabled on the switch to be used.
- Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.
- To use any of the AAA commands listed in this section or elsewhere, you must first enable AAA with the **aaa new-model** command.
- At a minimum, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting.
- The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined. A defined method list overrides the default method list.
- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Information About TACACS+

TACACS+ and Switch Access

This section describes TACACS+. TACACS+ provides detailed accounting information and flexible administrative control over the authentication and authorization processes. It is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.



Note Beginning with Cisco IOS Release 15.2(7)E3, the legacy command **tacacs-server** is deprecated. Use the **tacacs server** command if the software running on your device is Cisco IOS Release 15.2(7)E3 or later releases.

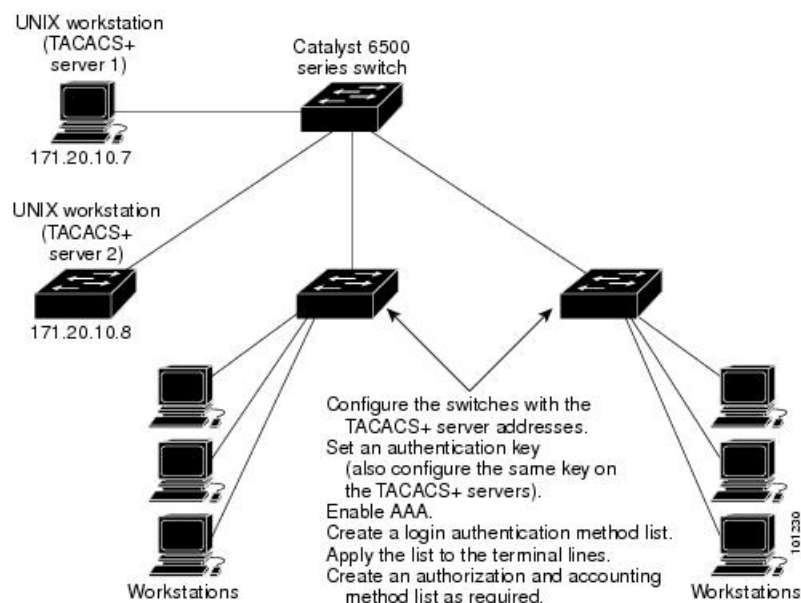
TACACS+ Overview

TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication, authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers.

Figure 89: Typical TACACS+ Network Configuration



TACACS+, administered through the AAA security services, can provide these services:

- **Authentication**—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name, service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- **Authorization**—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.
- **Accounting**—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or

to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:
 - **ACCEPT**—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
 - **REJECT**—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
 - **ERROR**—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an **ERROR** response is received, the switch typically tries to use an alternative method for authenticating the user.
 - **CONTINUE**—The user is prompted for additional authentication information.

After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an **ACCEPT** or **REJECT** authorization response. If an **ACCEPT** response is returned, the response contains data in the form of attributes that direct the **EXEC** or **NETWORK** session for that user and the services that the user can access:
 - Telnet, Secure Shell (SSH), rlogin, or privileged **EXEC** services
 - Connection parameters, including the host or client IP address, access list, and user timeouts

Method List

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list.

This process continues until there is successful communication with a listed method or the method list is exhausted.

If a method list is configured under VTY lines, the corresponding method list must be added to AAA. The following example shows how to configure a method list under a VTY line:

```
Device# configure terminal
Device(config)# line vty 0 4
Device(config)# authorization commands 15 auth1
```

The following example shows how to configure a method list in AAA:

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 auth1 group tacacs+
```

If no method list is configured under VTY lines, the default method list must be added to AAA. The following example shows a VTY configuration without a method list:

```
Device# configure terminal
Device(config)# line vty 0 4
```

The following example shows how to configure the default method list:

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 default group tacacs+
```

TACACS+ Configuration Options

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

TACACS+ Login Authentication

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Default TACACS+ Configuration

TACACS+ and AAA are disabled by default.

To prevent a lapse in security, you cannot configure TACACS+ through a network management application. When enabled, TACACS+ can authenticate users accessing the switch through the CLI.



Note Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

How to Configure Switch Access with TACACS+

This section describes how to configure your switch to support TACACS+.

Identifying the TACACS+ Server Host and Setting the Authentication Key

Follow these steps to identify the TACACS+ server host and set the authentication key:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **tacacs server *servername***
4. **aaa new-model**
5. **aaa group server tacacs+ *group-name***
6. **server *ip-address***
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> <code>enable</code>	
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	tacacs server <i>servername</i> Example: Switch(config)# <code>tacacs server yourserver</code>	Identifies the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them.
Step 4	aaa new-model Example: Switch(config)# <code>aaa new-model</code>	Enables AAA.
Step 5	aaa group server tacacs+ <i>group-name</i> Example: Switch(config)# <code>aaa group server tacacs+ your_server_group</code>	(Optional) Defines the AAA server-group with a group name. This command puts the Switch in a server group subconfiguration mode.
Step 6	server <i>ip-address</i> Example: Switch(config)# <code>server 10.1.2.3</code>	(Optional) Associates a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 3.
Step 7	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 8	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 9	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Configuring TACACS+ Login Authentication

Follow these steps to configure TACACS+ login authentication:

Before you begin

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports.



Note To secure the for HTTP access by using AAA methods, you must configure the with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the for HTTP access by using AAA methods.

For more information about the **ip http authentication** command, see the *Cisco IOS Security Command Reference, Release 12.4*.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login {default | list-name} method1 [method2...]**
5. **line [console | tty | vty] line-number [ending-line-number]**
6. **login authentication {default | list-name}**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	aaa new-model Example: <pre>Switch(config)# aaa new-model</pre>	Enables AAA.
Step 4	aaa authentication login {default list-name} method1 [method2...] Example: <pre>Switch(config)# aaa authentication login default tacacs+ local</pre>	<p>Creates a login authentication method list.</p> <ul style="list-style-type: none"> To create a default list that is used when a named list is <i>not</i> specified in the login authentication command, use the default keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. For <i>list-name</i>, specify a character string to name the list you are creating. For <i>method1...</i>, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. <p>Select one of these methods:</p> <ul style="list-style-type: none"> <i>enable</i>—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the enable password global configuration command. <i>group tacacs+</i>—Uses TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. <i>line</i>—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the password password line configuration command. <i>local</i>—Use the local username database for authentication. You must enter username information in the database. Use the username password global configuration command. <i>local-case</i>—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the username name password global configuration command. <i>none</i>—Do not use any authentication for login.
Step 5	line [console tty vty] line-number [ending-line-number] Example:	Enters line configuration mode, and configures the lines to which you want to apply the authentication list.

	Command or Action	Purpose
	Switch(config)# line 2 4	
Step 6	login authentication { default <i>list-name</i> } Example: Switch(config-line)# login authentication default	Applies the authentication list to a line or set of lines. <ul style="list-style-type: none"> • If you specify default, use the default list created with the aaa authentication login command. • For <i>list-name</i>, specify the list created with the aaa authentication login command.
Step 7	end Example: Switch(config-line)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Switch# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.



Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to specify TACACS+ authorization for privileged EXEC access and network services:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network tacacs+**
4. **aaa authorization exec tacacs+**

5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>aaa authorization network tacacs+</p> <p>Example:</p> <pre>Switch(config)# aaa authorization network tacacs+</pre>	<p>Configures the switch for user TACACS+ authorization for all network-related service requests.</p>
Step 4	<p>aaa authorization exec tacacs+</p> <p>Example:</p> <pre>Switch(config)# aaa authorization exec tacacs+</pre>	<p>Configures the switch for user TACACS+ authorization if the user has privileged EXEC access.</p> <p>The exec keyword might return user profile information (such as autocommand information).</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Starting TACACS+ Accounting

Follow these steps to start TACACS+ Accounting:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop tacacs+**
4. **aaa accounting exec start-stop tacacs+**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	aaa accounting network start-stop tacacs+ Example: <pre>Switch(config)# aaa accounting network start-stop tacacs+</pre>	Enables TACACS+ accounting for all network-related service requests.
Step 4	aaa accounting exec start-stop tacacs+ Example: <pre>Switch(config)# aaa accounting exec start-stop tacacs+</pre>	Enables TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To establish a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Establishing a Session with a Router if the AAA Server is Unreachable

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Monitoring TACACS+

Table 117: Commands for Displaying TACACS+ Information

Command	Purpose
show tacacs	Displays TACACS+ server statistics.



CHAPTER 54

Configuring RADIUS

- [Finding Feature Information, on page 1099](#)
- [Prerequisites for Configuring RADIUS, on page 1099](#)
- [Restrictions for Configuring RADIUS, on page 1100](#)
- [Information about RADIUS, on page 1100](#)
- [How to Configure RADIUS, on page 1121](#)
- [Monitoring CoA Functionality, on page 1137](#)
- [Configuration Examples for Controlling Switch Access with RADIUS, on page 1138](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Configuring RADIUS

This section lists the prerequisites for controlling Switch access with RADIUS.

General:

- RADIUS and Authentication, Authorization, and Accounting (AAA) must be enabled to use any of the configuration commands in this chapter.
- RADIUS is facilitated through AAA and can be enabled only through AAA commands.
- Use the **aaa new-model** global configuration command to enable AAA.
- Use the **aaa authentication** global configuration command to define method lists for RADIUS authentication.
- Use **line** and **interface** commands to enable the defined method lists to be used.

- At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.
- You should have access to and should configure a RADIUS server before configuring RADIUS features on your Switch.
- The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server Version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.
- To use the Change-of-Authorization (CoA) interface, a session must already exist on the switch. CoA can be used to identify a session and enforce a disconnect request. The update affects only the specified session.

For RADIUS operation:

- Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled.

Restrictions for Configuring RADIUS

This topic covers restrictions for controlling Switch access with RADIUS.

General:

- To prevent a lapse in security, you cannot configure RADIUS through a network management application.

RADIUS is not suitable in the following network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Information about RADIUS

RADIUS and Switch Access

This section describes how to enable and configure RADIUS. RADIUS provides detailed accounting information and flexible administrative control over the authentication and authorization processes.

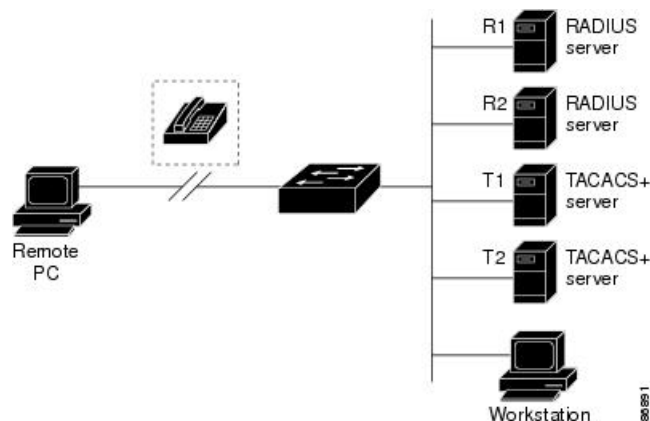
RADIUS Overview

RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information.

Use RADIUS in these network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a *smart card* access control system. In one case, RADIUS has been used with Enigma's security cards to validate users and to grant access to network resources.
- Networks already using RADIUS. You can add a Cisco Switch containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server. See Figure: Transitioning from RADIUS to TACACS+ Services below.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network through a protocol such as IEEE 802.1x. For more information about this protocol, see *Configuring IEEE 802.1x Port-Based Authentication* chapter.
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

Figure 90: Transitioning from RADIUS to TACACS+ Services



RADIUS Operation

When a user attempts to log in and authenticate to a Switch that is access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of the following responses from the RADIUS server:
 - ACCEPT—The user is authenticated.
 - REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.
 - CHALLENGE—A challenge requires additional data from the user.
 - CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

RADIUS Change of Authorization

The RADIUS Change of Authorization (CoA) provides a mechanism to change the attributes of an authentication, authorization, and accounting (AAA) session after it is authenticated. When a policy changes for a user or user group in AAA, administrators can send RADIUS CoA packets from the AAA server such as a Cisco Secure Access Control Server (ACS) to reinitialize authentication and apply the new policy. This section provides an overview of the RADIUS interface including available primitives and how they are used during a CoA.

- Change-of-Authorization Requests
- CoA Request Response Code
- CoA Request Commands
- Session Reauthentication
- Stacking Guidelines for Session Termination

A standard RADIUS interface is typically used in a pulled model where the request originates from a network attached device and the response come from the queried servers. Catalyst support the RADIUS CoA extensions defined in RFC 5176 that are typically used in a pushed model and allow for the dynamic reconfiguring of sessions from external AAA or policy servers.

The supports these per-session CoA requests:

- Session reauthentication
- Session termination
- Session termination with port shutdown
- Session termination with port bounce

This feature is integrated with Cisco Secure Access Control Server (ACS) 5.1.

The RADIUS interface is enabled by default on Catalyst . However, some basic configuration is required for the following attributes:

- Security and Password—refer to the “Preventing Unauthorized Access to Your Switch” section in this guide.
- Accounting—refer to the “Starting RADIUS Accounting” section in the Configuring Switch-Based Authentication chapter in this guide.

Cisco IOS software supports the RADIUS CoA extensions defined in RFC 5176 that are typically used in a push model to allow the dynamic reconfiguring of sessions from external AAA or policy servers. Per-session CoA requests are supported for session identification, session termination, host reauthentication, port shutdown, and port bounce. This model comprises one request (CoA-Request) and two possible response codes:

- CoA acknowledgement (ACK) [CoA-ACK]
- CoA nonacknowledgement (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a AAA or policy server) and directed to the device that acts as a listener.

The table below shows the RADIUS CoA commands and vendor-specific attributes (VSAs) supported by Identity-Based Networking Services. All CoA commands must include the session identifier between the device and the CoA client.

Table 118: RADIUS CoA Commands Supported by Identity-Based Networking Services

CoA Command	Cisco VSA
Activate service	Cisco:Avpair="subscriber:command=activate-service" Cisco:Avpair="subscriber:service-name=<service-name>" Cisco:Avpair="subscriber:precedence=<precedence-number>" Cisco:Avpair="subscriber:activation-mode=replace-all"
Deactivate service	Cisco:Avpair="subscriber:command=deactivate-service" Cisco:Avpair="subscriber:service-name=<service-name>"
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"
Session query	Cisco:Avpair="subscriber:command=session-query"
Session reauthenticate	Cisco:Avpair="subscriber:command=reauthenticate" Cisco:Avpair="subscriber:reauthenticate-type=last" or Cisco:Avpair="subscriber:reauthenticate-type=rerun"
Session terminate	This is a standard disconnect request and does not require a VSA.
Interface template	Cisco:AVpair="interface-template-name=<interfacetemplate>"

Change-of-Authorization Requests

Change of Authorization (CoA) requests, as described in RFC 5176, are used in a push model to allow for session identification, host reauthentication, and session termination. The model is comprised of one request (CoA-Request) and two possible response codes:

- CoA acknowledgment (ACK) [CoA-ACK]
- CoA non-acknowledgment (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a RADIUS or policy server) and directed to the switch that acts as a listener.

RFC 5176 Compliance

The Disconnect Request message, which is also referred to as Packet of Disconnect (POD), is supported by the switch for session termination.

This table shows the IETF attributes supported for this feature.

Table 119: Supported IETF Attributes

Attribute Number	Attribute Name
24	State
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

This table shows the possible values for the Error-Cause attribute.

Table 120: Error-Cause Values

Value	Explanation
201	Residual Session Context Removed
202	Invalid EAP Packet (Ignored)
401	Unsupported Attribute
402	Missing Attribute
403	NAS Identification Mismatch
404	Invalid Request
405	Unsupported Service
406	Unsupported Extension

Value	Explanation
407	Invalid Attribute Value
501	Administratively Prohibited
502	Request Not Routable (Proxy)
503	Session Context Not Found
504	Session Context Not Removable
505	Other Proxy Processing Error
506	Resources Unavailable
507	Request Initiated
508	Multiple Session Selection Unsupported

CoA Request Response Code

The CoA Request response code can be used to convey a command to the switch.

The packet format for a CoA Request Response code as defined in RFC 5176 consists of the following fields: Code, Identifier, Length, Authenticator, and Attributes in the Type:Length:Value (TLV) format. The Attributes field is used to carry Cisco vendor-specific attributes (VSAs).

Session Identification

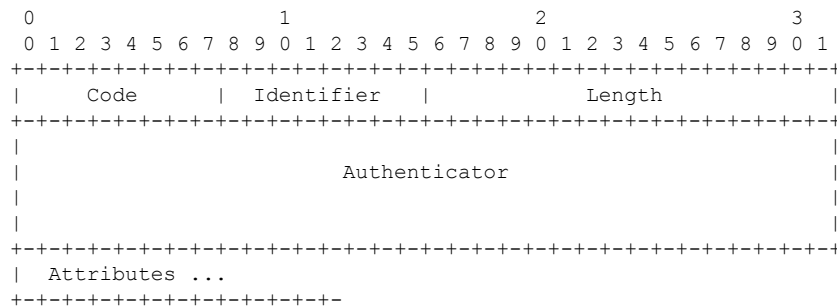
For disconnect and CoA requests targeted at a particular session, the switch locates the session based on one or more of the following attributes:

- Acct-Session-Id (IETF attribute #44)
- Audit-Session-Id (Cisco VSA)
- Calling-Station-Id (IETF attribute #31 which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
 - Framed-IPv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
 - Framed-IPv6-Address
- Plain IP Address (IETF attribute #8)

Unless all session identification attributes included in the CoA message match the session, the switch returns a Disconnect-NAK or CoA-NAK with the “Invalid Attribute Value” error-code attribute.

If more than one session identification attribute is included in the message, all the attributes must match the session or the switch returns a Disconnect- negative acknowledgment (NAK) or CoA-NAK with the error code “Invalid Attribute Value.”

The packet format for a CoA Request code as defined in RFC 5176 consists of the fields: Code, Identifier, Length, Authenticator, and Attributes in Type:Length:Value (TLV) format.



The attributes field is used to carry Cisco vendor-specific attributes (VSAs).

For CoA requests targeted at a particular enforcement policy, the device returns a CoA-NAK with the error code “Invalid Attribute Value” if any of the above session identification attributes are included in the message.

CoA ACK Response Code

If the authorization state is changed successfully, a positive acknowledgment (ACK) is sent. The attributes returned within CoA ACK will vary based on the CoA Request and are discussed in individual CoA Commands.

CoA NAK Response Code

A negative acknowledgment (NAK) indicates a failure to change the authorization state and can include attributes that indicate the reason for the failure. Use **show** commands to verify a successful CoA.

CoA Request Commands

Table 121: CoA Commands Supported on the

Command	Cisco VSA
8	
Reauthenticate host	Cisco:Avpair=“subscriber:command=reauthenticate”
Terminate session	This is a standard disconnect request that does not require a VSA.
Bounce host port	Cisco:Avpair=“subscriber:command=bounce-host-port”
Disable host port	Cisco:Avpair=“subscriber:command=disable-host-port”

⁸ All CoA commands must include the session identifier between the and the CoA client.

Session Reauthentication

The AAA server typically generates a session reauthentication request when a host with an unknown identity or posture joins the network and is associated with a restricted access authorization profile (such as a guest VLAN). A reauthentication request allows the host to be placed in the appropriate authorization group when its credentials are known.

To initiate session authentication, the AAA server sends a standard CoA-Request message which contains a Cisco VSA in this form: *Cisco:Avpair=“subscriber:command=reauthenticate”* and one or more session identification attributes.

The current session state determines the switch response to the message. If the session is currently authenticated by IEEE 802.1x, the switch responds by sending an EAPoL (Extensible Authentication Protocol over Lan) -RequestId message to the server.

If the session is currently authenticated by MAC authentication bypass (MAB), the switch sends an access-request to the server, passing the same identity attributes used for the initial successful authentication.

If session authentication is in progress when the switch receives the command, the switch terminates the process, and restarts the authentication sequence, starting with the method configured to be attempted first.

If the session is not yet authorized, or is authorized via guest VLAN, or critical VLAN, or similar policies, the reauthentication message restarts the access control methods, beginning with the method configured to be attempted first. The current authorization of the session is maintained until the reauthentication leads to a different authorization result.

Session Termination

There are three types of CoA requests that can trigger session termination. A CoA Disconnect-Request terminates the session, without disabling the host port. This command causes re-initialization of the authenticator state machine for the specified host, but does not restrict that host access to the network.

To restrict a host's access to the network, use a CoA Request with the `Cisco:Avpair="subscriber:command=disable-host-port" VSA`. This command is useful when a host is known to be causing problems on the network, and you need to immediately block network access for the host. When you want to restore network access on the port, re-enable it using a non-RADIUS mechanism.

When a device with no supplicant, such as a printer, needs to acquire a new IP address (for example, after a VLAN change), terminate the session on the host port with port-bounce (temporarily disable and then re-enable the port).

CoA Disconnect-Request

This command is a standard Disconnect-Request. If the session cannot be located, the switch returns a Disconnect-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch terminates the session. After the session has been completely removed, the switch returns a Disconnect-ACK.

If the switch fails-over to a standby switch before returning a Disconnect-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the session is not found following re-sending, a Disconnect-ACK is sent with the "Session Context Not Found" error-code attribute.

CoA Request: Disable Host Port

The RADIUS server CoA disable port command administratively shuts down the authentication port that is hosting a session, resulting in session termination. This command is useful when a host is known to cause problems on the network and network access needs to be immediately blocked for the host. To restore network access on the port, reenable it using a non-RADIUS mechanism. This command is carried in a standard CoA-Request message that has this new vendor-specific attribute (VSA):

```
Cisco:Avpair="subscriber:command=disable-host-port"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the "Session Identification" section. If the session cannot be located, the switch returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch disables the hosting port and returns a CoA-ACK message.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is restarted on the new active switch.



Note A Disconnect-Request failure following command re-sending could be the result of either a successful session termination before change-over (if the Disconnect-ACK was not sent) or a session termination by other means (for example, a link failure) that occurred after the original command was issued and before the standby switch became active.

CoA Request: Bounce-Port

A RADIUS server CoA bounce port sent from a RADIUS server can cause a link flap on an authentication port, which triggers DHCP renegotiation from one or more hosts connected to this port. This incident can occur when there is a VLAN change and the endpoint is a device (such as a printer) that does not have a mechanism to detect a change on this authentication port. The CoA bounce port is carried in a standard CoA-Request message that contains the following VSA:

```
Cisco:Avpair="subscriber:command=bounce-host-port"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes. If the session cannot be located, the switch returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the switch disables the hosting port for a period of 10 seconds, re-enables it (port-bounce), and returns a CoA-ACK.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is re-started on the new active switch.

Default RADIUS Configuration

RADIUS and AAA are disabled by default.

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP

port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the %RADIUS-4-RADIUS_DEAD message appears, and then the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings.

RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list. The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to different UDP ports on a server at the same IP address. If you configure two different host entries on the same RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a fail-over backup to the first one. If the first host entry fails to provide accounting services, the network access server tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order in which they are configured.)

AAA Authorization

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

RADIUS Accounting

The AAA accounting feature tracks the services that users are using and the amount of network resources that they are consuming. When you enable AAA accounting, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. You can then analyze the data for network management, client billing, or auditing.

Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the switch and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attributevalue (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, the following AV pair causes Cisco's "multiple named IP address pools" feature to be activated during IP authorization (during PPP's Internet Protocol Control Protocol (IPCP) address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

If you insert an "*", the AV pair "ip:addr-pool=first" becomes optional. Note that any AV pair can be made optional:

```
cisco-avpair= "ip:addr-pool*first"
```

The following example shows how to cause a user logging in from a network access server to have immediate access to EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

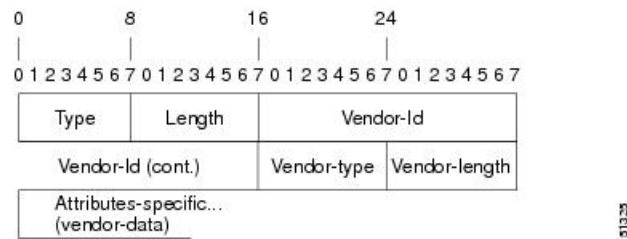
Attribute 26 contains the following three elements:

- Type

- Length
- String (also known as data)
 - Vendor-Id
 - Vendor-Type
 - Vendor-Length
 - Vendor-Data

The figure below shows the packet format for a VSA encapsulated “behind” attribute 26.

Figure 91: VSA Encapsulated Behind Attribute 26



Note It is up to the vendor to specify the format of their VSA. The Attribute-Specific field (also known as Vendor-Data) is dependent on the vendor's definition of that attribute.

The table below describes significant fields listed in the Vendor-Specific RADIUS IETF Attributes table (second table below), which lists supported vendor-specific RADIUS attributes (IETF attribute 26).

Table 122: Vendor-Specific Attributes Table Field Descriptions

Field	Description
Number	All attributes listed in the following table are extensions of IETF attribute 26.
Vendor-Specific Command Codes	A defined code used to identify a particular vendor. Code 9 defines Cisco VSAs, 311 defines Microsoft VSAs, and 529 defines Ascend VSAs.
Sub-Type Number	The attribute ID number. This number is much like the ID numbers of IETF attributes, except it is a “second layer” ID number encapsulated behind attribute 26.
Attribute	The ASCII string name of the attribute.
Description	Description of the attribute.

Table 123: Vendor-Specific RADIUS IETF Attributes

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
MS-CHAP Attributes				

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	311	1	MSCHAP-Response	Contains the response value provided by a PPP MS-CHAP user in response to the challenge. It is only used in Access-Request packets. This attribute is identical to the PPP CHAP Identifier. (RFC 2548
26	311	11	MSCHAP-Challenge	Contains the challenge sent by a network access server to an MS-CHAP user. It can be used in both Access-Request and Access-Challenge packets. (RFC 2548)
VPDN Attributes				
26	9	1	l2tp-cm-local-window-size	Specifies the maximum receive window size for L2TP control messages. This value is advertised to the peer during tunnel establishment.
26	9	1	l2tp-drop-out-of-order	Respects sequence numbers on data packets by dropping those that are received out of order. This does not ensure that sequence numbers will be sent on data packets, just how to handle them if they are received.
26	9	1	l2tp-hello-interval	Specifies the number of seconds for the hello keepalive interval. Hello packets are sent when no data has been sent on a tunnel for the number of seconds configured here.
26	9	1	l2tp-hidden-avp	When enabled, sensitive AVPs in L2TP control messages are scrambled or hidden.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	l2tp-nosession-timeout	Specifies the number of seconds that a tunnel will stay active with no sessions before timing out and shutting down.
26	9	1	tunnel-tos-reflect	Copies the IP ToS field from the IP header of each payload packet to the IP header of the tunnel packet for packets entering the tunnel at the LNS.
26	9	1	l2tp-tunnel-authen	If this attribute is set, it performs L2TP tunnel authentication.
26	9	1	l2tp-tunnel-password	Shared secret used for L2TP tunnel authentication and AVP hiding.
26	9	1	l2tp-udp-checksum	This is an authorization attribute and defines whether L2TP should perform UDP checksums for data packets. Valid values are "yes" and "no." The default is no.
Store and Forward Fax Attributes				
26	9	3	Fax-Account-Id-Origin	Indicates the account ID origin as defined by system administrator for the mmoip aaa receive-id or the mmoip aaa send-id commands.
26	9	4	Fax-Msg-Id=	Indicates a unique fax message identification number assigned by Store and Forward Fax.
26	9	5	Fax-Pages	Indicates the number of pages transmitted or received during this fax session. This page count includes cover pages.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	6	Fax-Coverpage-Flag	Indicates whether or not a cover page was generated by the off-ramp gateway for this fax session. True indicates that a cover page was generated; false means that a cover page was not generated.
26	9	7	Fax-Modem-Time	Indicates the amount of time in seconds the modem sent fax data (x) and the amount of time in seconds of the total fax session (y), which includes both fax-mail and PSTN time, in the form x/y. For example, 10/15 means that the transfer time took 10 seconds, and the total fax session took 15 seconds.
26	9	8	Fax-Connect-Speed	Indicates the modem speed at which this fax-mail was initially transmitted or received. Possible values are 1200, 4800, 9600, and 14400.
26	9	9	Fax-Recipient-Count	Indicates the number of recipients for this fax transmission. Until e-mail servers support Session mode, the number should be 1.
26	9	10	Fax-Process-Abort-Flag	Indicates that the fax session was cancelled or successful. True means that the session was cancelled; false means that the session was successful.
26	9	11	Fax-Dsn-Address	Indicates the address to which DSNs will be sent.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	12	Fax-Dsn-Flag	Indicates whether or not DSN has been enabled. True indicates that DSN has been enabled; false means that DSN has not been enabled.
26	9	13	Fax-Mdn-Address	Indicates the address to which MDNs will be sent.
26	9	14	Fax-Mdn-Flag	Indicates whether or not message delivery notification (MDN) has been enabled. True indicates that MDN had been enabled; false means that MDN had not been enabled.
26	9	15	Fax-Auth-Status	Indicates whether or not authentication for this fax session was successful. Possible values for this field are success, failed, bypassed, or unknown.
26	9	16	Email-Server-Address	Indicates the IP address of the e-mail server handling the on-ramp fax-mail message.
26	9	17	Email-Server-Ack-Flag	Indicates that the on-ramp gateway has received a positive acknowledgment from the e-mail server accepting the fax-mail message.
26	9	18	Gateway-Id	Indicates the name of the gateway that processed the fax session. The name appears in the following format: hostname.domain-name.
26	9	19	Call-Type	Describes the type of fax activity: fax receive or fax send.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	20	Port-Used	Indicates the slot/port number of the Cisco AS5300 used to either transmit or receive this fax-mail.
26	9	21	Abort-Cause	If the fax session cancels, indicates the system component that signaled the cancel operation. Examples of system components that could trigger a cancel operation are FAP (Fax Application Process), TIFF (the TIFF reader or the TIFF writer), fax-mail client, fax-mail server, ESMTP client, or ESMTP server.
H323 Attributes				
26	9	23	Remote-Gateway-ID (h323-remote-address)	Indicates the IP address of the remote gateway.
26	9	24	Connection-ID (h323-conf-id)	Identifies the conference ID.
26	9	25	Setup-Time (h323-setup-time)	Indicates the setup time for this connection in Coordinated Universal Time (UTC) formerly known as Greenwich Mean Time (GMT) and Zulu time.
26	9	26	Call-Origin (h323-call-origin)	Indicates the origin of the call relative to the gateway. Possible values are originating and terminating (answer).
26	9	27	Call-Type (h323-call-type)	Indicates call leg type. Possible values are telephony and VoIP .
26	9	28	Connect-Time (h323-connect-time)	Indicates the connection time for this call leg in UTC.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	29	Disconnect-Time (h323-disconnect-time)	Indicates the time this call leg was disconnected in UTC.
26	9	30	Disconnect-Cause (h323-disconnect-cause)	Specifies the reason a connection was taken offline per Q.931 specification.
26	9	31	Voice-Quality (h323-voice-quality)	Specifies the impairment factor (ICPIF) affecting voice quality for a call.
26	9	33	Gateway-ID (h323-gw-id)	Indicates the name of the underlying gateway.
Large Scale Dialout Attributes				
26	9	1	callback-dialstring	Defines a dialing string to be used for callback.
26	9	1	data-service	No description available.
26	9	1	dial-number	Defines the number to dial.
26	9	1	force-56	Determines whether the network access server uses only the 56 K portion of a channel, even when all 64 K appear to be available.
26	9	1	map-class	Allows the user profile to reference information configured in a map class of the same name on the network access server that dials out.
26	9	1	send-auth	Defines the protocol to use (PAP or CHAP) for username-password authentication following CLID authentication.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-name	<p>PPP name authentication. To apply for PAP, do not configure the ppp pap sent-name password command on the interface. For PAP, “preauth:send-name” and “preauth:send-secret” will be used as the PAP username and PAP password for outbound authentication. For CHAP, “preauth:send-name” will be used not only for outbound authentication, but also for inbound authentication. For a CHAP inbound case, the NAS will use the name defined in “preauth:send-name” in the challenge packet to the caller box.</p> <p>Note The send-name attribute has changed over time: Initially, it performed the functions now provided by both the send-name and remote-name attributes. Because the remote-name attribute has been added, the send-name attribute is restricted to its current behavior.</p>

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-secret	PPP password authentication. The vendor-specific attributes (VSAs) "preauth:send-name" and "preauth:send-secret" will be used as the PAP username and PAP password for outbound authentication. For a CHAP outbound case, both "preauth:send-name" and "preauth:send-secret" will be used in the response packet.
26	9	1	remote-name	Provides the name of the remote host for use in large-scale dial-out. Dialer checks that the large-scale dial-out remote name matches the authenticated name, to protect against accidental user RADIUS misconfiguration. (For example, dialing a valid phone number but connecting to the wrong device.)
Miscellaneous Attributes				

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	2	Cisco-NAS-Port	<p>Specifies additional vendor specific attribute (VSA) information for NAS-Port accounting. To specify additional NAS-Port information in the form an Attribute-Value Pair (AVPair) string, use the radius-server vsa send global configuration command.</p> <p>Note This VSA is typically used in Accounting, but may also be used in Authentication (Access-Request) packets.</p>
26	9	1	min-links	Sets the minimum number of links for MLP.
26	9	1	proxyacl#<n>	Allows users to configure the downloadable user profiles (dynamic ACLs) by using the authentication proxy feature so that users can have the configured authorization to permit traffic going through the configured interfaces.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	spi	Carries the authentication information needed by the home agent to authenticate a mobile node during registration. The information is in the same syntax as the ip mobile secure host <addr> configuration command. Basically it contains the rest of the configuration command that follows that string, verbatim. It provides the Security Parameter Index (SPI), key, authentication algorithm, authentication mode, and replay protection timestamp range.

Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the switch and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius server** global configuration commands.

How to Configure RADIUS

Identifying the RADIUS Server Host

To apply these settings globally to all RADIUS servers communicating with the Switch, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**.

You can configure the Switch to use AAA server groups to group existing server hosts for authentication. For more information, see Related Topics below.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the Switch and the key string to be shared by both the server and the Switch. For more information, see the RADIUS server documentation.

Follow these steps to configure per-server RADIUS server communication.

Before you begin

If you configure both global and per-server functions (timeout, retransmission, and key commands) on the device, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands. For information on configuring these settings on all RADIUS servers, see Related Topics below.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** *{hostname | ip-address}* [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	radius-server host <i>{hostname ip-address}</i> [auth-port <i>port-number</i>] [acct-port <i>port-number</i>] [timeout <i>seconds</i>] [retransmit <i>retries</i>] [key <i>string</i>] Example: <pre>Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1</pre>	Specifies the IP address or hostname of the remote RADIUS server host. <ul style="list-style-type: none"> • (Optional) For auth-port <i>port-number</i>, specify the UDP destination port for authentication requests. • (Optional) For acct-port <i>port-number</i>, specify the UDP destination port for accounting requests. • (Optional) For timeout <i>seconds</i>, specify the time interval that the Switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the radius-server timeout global configuration command setting. If no timeout is set with the radius-server host command, the setting of the radius-server timeout command is used. • (Optional) For retransmit <i>retries</i>, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The

	Command or Action	Purpose
		<p>range is 1 to 1000. If no retransmit value is set with the radius-server host command, the setting of the radius-server retransmit global configuration command is used.</p> <ul style="list-style-type: none"> • (Optional) For key string, specify the authentication and encryption key used between the Switch and the RADIUS daemon running on the RADIUS server. <p>Note The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the radius-server host command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.</p> <p>To configure the Switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The Switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring RADIUS Login Authentication

Follow these steps to configure RADIUS login authentication:

Before you begin

To secure the device for HTTP access by using AAA methods, you must configure the device with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the device for HTTP access by using AAA methods.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login {default | list-name} method1 [method2...]**
5. **line [console | tty | vty] line-number [ending-line-number]**
6. **login authentication {default | list-name}**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	aaa new-model Example: <pre>Switch(config)# aaa new-model</pre>	Enables AAA.
Step 4	aaa authentication login {default list-name} method1 [method2...] Example: <pre>Switch(config)# aaa authentication login default local</pre>	Creates a login authentication method list. <ul style="list-style-type: none"> • To create a default list that is used when a named list is <i>not</i> specified in the login authentication command, use the default keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For <i>list-name</i>, specify a character string to name the list you are creating.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>method1...</i>, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. <p>Select one of these methods:</p> <ul style="list-style-type: none"> <i>enable</i>—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the enable password global configuration command. <i>group radius</i>—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. <i>line</i>—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the password password line configuration command. <i>local</i>—Use the local username database for authentication. You must enter username information in the database. Use the username name password global configuration command. <i>local-case</i>—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the username password global configuration command. <i>none</i>—Do not use any authentication for login.
Step 5	line [console tty vty] <i>line-number</i> [<i>ending-line-number</i>] Example: <pre>Switch(config)# line 1 4</pre>	Enters line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 6	login authentication {default <i>list-name</i> } Example: <pre>Switch(config)# login authentication default</pre>	<p>Applies the authentication list to a line or set of lines.</p> <ul style="list-style-type: none"> If you specify default, use the default list created with the aaa authentication login command. For <i>list-name</i>, specify the list created with the aaa authentication login command.
Step 7	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 8	show running-config Example: Switch# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Defining AAA Server Groups

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

Follow these steps to define AAA server groups:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius server** *name*
4. **address** {**ipv4** | **ipv6**} {*ip-address* | *hostname*} **auth-port** *port-number* **acct-port** *port-number*
5. **key** *string*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	radius server <i>name</i> Example: Switch(config)# <code>radius server ISE</code>	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode. The device also supports RADIUS for IPv6.
Step 4	address { ipv4 ipv6 } { <i>ip-address</i> <i>hostname</i> } auth-port <i>port-number</i> acct-port <i>port-number</i> Example: Switch(config-radius-server)# <code>address ipv4 10.1.1.1 auth-port 1645 acct-port 1646</code>	Configures the IPv4 address for the RADIUS server accounting and authentication parameters.
Step 5	key <i>string</i> Example: Switch(config-radius-server)# <code>key cisco123</code>	Specifies the authentication and encryption key for all RADIUS communications between the device and the RADIUS server.
Step 6	end Example: Switch(config-radius-server)# <code>end</code>	Exits RADIUS server configuration mode and returns to privileged EXEC mode.
Step 7	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 8	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring RADIUS Authorization for User Privileged Access and Network Services



Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to configure RADIUS authorization for user privileged access and network services:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network radius**
4. **aaa authorization exec radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	aaa authorization network radius Example: Switch(config)# aaa authorization network radius	Configures the device for user RADIUS authorization for all network-related service requests.
Step 4	aaa authorization exec radius Example: Switch(config)# aaa authorization exec radius	Configures the device for user RADIUS authorization if the user has privileged EXEC access. The exec keyword might return user profile information (such as autocommand information).

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Starting RADIUS Accounting

Follow these steps to start RADIUS accounting:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop radius**
4. **aaa accounting exec start-stop radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> <code>enable</code>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	aaa accounting network start-stop radius Example: Switch(config)# <code>aaa accounting network start-stop radius</code>	Enables RADIUS accounting for all network-related service requests.
Step 4	aaa accounting exec start-stop radius Example: Switch(config)# <code>aaa accounting exec start-stop radius</code>	Enables RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure settings for all RADIUS servers:

SUMMARY STEPS

1. `configure terminal`

2. **radius-server key** *string*
3. **radius-server retransmit** *retries*
4. **radius-server timeout** *seconds*
5. **radius-server deadtime** *minutes*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	radius-server key <i>string</i> Example: <pre>Switch(config)# radius-server key your_server_key</pre> <pre>Switch(config)# key your_server_key</pre>	Specifies the shared secret text string used between the switch and all RADIUS servers. Note The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 3	radius-server retransmit <i>retries</i> Example: <pre>Switch(config)# radius-server retransmit 5</pre>	Specifies the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	radius-server timeout <i>seconds</i> Example: <pre>Switch(config)# radius-server timeout 3</pre>	Specifies the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.
Step 5	radius-server deadtime <i>minutes</i> Example: <pre>Switch(config)# radius-server deadtime 0</pre>	When a RADIUS server is not responding to authentication requests, this command specifies a time to stop the request on that server. This avoids the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
Step 6	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 7	show running-config Example: Switch# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Device to Use Vendor-Specific RADIUS Attributes

Follow these steps to configure the device to use vendor-specific RADIUS attributes:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server vsa send [accounting | authentication]**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	radius-server vsa send [accounting authentication] Example:	Enables the device to recognize and use VSAs as defined by RADIUS IETF attribute 26.

	Command or Action	Purpose
	Switch(config)# radius-server vsa send accounting	<ul style="list-style-type: none"> • (Optional) Use the accounting keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the authentication keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. <p>If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.</p>
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Device for Vendor-Proprietary RADIUS Server Communication

Follow these steps to configure the device to use vendor-proprietary RADIUS server communication:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host {hostname | ip-address} non-standard**
4. **radius-server key string**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	radius-server host {hostname ip-address} non-standard Example: <pre>Switch(config)# radius-server host 172.20.30.15 non-standard</pre>	Specifies the IP address or hostname of the remote RADIUS server host and identifies that it is using a vendor-proprietary implementation of RADIUS.
Step 4	radius-server key string Example: <pre>Switch(config)# radius-server key rad124</pre>	Specifies the shared secret text string used between the device and the vendor-proprietary RADIUS server. The device and the RADIUS server use this text string to encrypt passwords and exchange responses. <p>Note The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.</p>
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Configuring CoA on the Device

Follow these steps to configure CoA on a device. This procedure is required.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `aaa new-model`
4. `aaa server radius dynamic-author`
5. `client {ip-address | name} [vrf vrfname] [server-key string]`
6. `server-key [0 | 7] string`
7. `port port-number`
8. `auth-type {any | all | session-key}`
9. `ignore session-key`
10. `ignore server-key`
11. `authentication command bounce-port ignore`
12. `authentication command disable-port ignore`
13. `end`
14. `show running-config`
15. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>aaa new-model</code></p> <p>Example:</p> <pre>Switch(config)# aaa new-model</pre>	<p>Enables AAA.</p>

	Command or Action	Purpose
Step 4	aaa server radius dynamic-author Example: <pre>Switch(config)# aaa server radius dynamic-author</pre>	Configures the device as an authentication, authorization, and accounting (AAA) server to facilitate interaction with an external policy server.
Step 5	client { <i>ip-address</i> <i>name</i> } [vrf <i>vrfname</i>] [server-key <i>string</i>] 	Enters dynamic authorization local server configuration mode and specifies a RADIUS client from which a device will accept CoA and disconnect requests.
Step 6	server-key [0 7] <i>string</i> Example: <pre>Switch(config-sg-radius)# server-key your_server_key</pre>	Configures the RADIUS key to be shared between a device and RADIUS clients.
Step 7	port <i>port-number</i> Example: <pre>Switch(config-sg-radius)# port 25</pre>	Specifies the port on which a device listens for RADIUS requests from configured RADIUS clients.
Step 8	auth-type { <i>any</i> <i>all</i> <i>session-key</i> } Example: <pre>Switch(config-sg-radius)# auth-type any</pre>	<p>Specifies the type of authorization the device uses for RADIUS clients.</p> <p>The client must match all the configured attributes for authorization.</p>
Step 9	ignore session-key	(Optional) Configures the device to ignore the session-key. For more information about the ignore command, see the <i>Cisco IOS Intelligent Services Gateway Command Reference</i> on Cisco.com.
Step 10	ignore server-key Example: <pre>Switch(config-sg-radius)# ignore server-key</pre>	(Optional) Configures the device to ignore the server-key. For more information about the ignore command, see the <i>Cisco IOS Intelligent Services Gateway Command Reference</i> on Cisco.com.
Step 11	authentication command bounce-port ignore Example: <pre>Switch(config-sg-radius)# authentication command bounce-port ignore</pre>	(Optional) Configures the device to ignore a CoA request to temporarily disable the port hosting a session. The purpose of temporarily disabling the port is to trigger a DHCP renegotiation from the host when a VLAN change occurs and there is no supplicant on the endpoint to detect the change.
Step 12	authentication command disable-port ignore Example:	(Optional) Configures the device to ignore a nonstandard command requesting that the port hosting a session be

	Command or Action	Purpose
	Switch(config-sg-radius)# authentication command disable-port ignore	administratively shut down. Shutting down the port results in termination of the session. Use standard CLI or SNMP commands to re-enable the port.
Step 13	end Example: Switch(config-sg-radius)# end	Returns to privileged EXEC mode.
Step 14	show running-config Example: Switch# show running-config	Verifies your entries.
Step 15	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring CoA Functionality

Table 124: Privileged EXEC show Commands

Command	Purpose
show aaa attributes protocol radius	Displays AAA attributes of RADIUS commands.

Table 125: Global Troubleshooting Commands

Command	Purpose
debug radius	Displays information for troubleshooting RADIUS.
debug aaa coa	Displays information for troubleshooting CoA processing.
debug aaa pod	Displays information for troubleshooting POD packets.
debug aaa subsys	Displays information for troubleshooting POD packets.
debug cmdhd [detail error events]	Displays information for troubleshooting command headers.

For detailed information about the fields in these displays, see the command reference for this release.

Configuration Examples for Controlling Switch Access with RADIUS

Examples: Identifying the RADIUS Server Host

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Switch(config)# radius-server host host1
```

Example: Using Two Different RADIUS Group Servers

In this example, the switch is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

Examples: Configuring the Switch to Use Vendor-Specific RADIUS Attributes

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during PPP IPCP address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

This example shows how to provide a user logging in from a switch with immediate access to privileged EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= "tunnel-type (#64)=VLAN(13) "  
cisco-avpair= "tunnel-medium-type (#65)=802 media (6) "  
cisco-avpair= "tunnel-private-group-id (#81)=vlanid"
```

This example shows how to apply an input ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"  
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"  
cisco-avpair= "mac:inacl#3=deny any any deernet-iv"
```

This example shows how to apply an output ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

Example: Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the switch and the server:

```
Switch(config)# radius-server host 172.20.30.15 nonstandard  
Switch(config)# radius-server key rad124
```




CHAPTER 55

Configuring Kerberos

- [Prerequisites for Controlling Switch Access with Kerberos, on page 1141](#)
- [Information about Kerberos, on page 1141](#)
- [How to Configure Kerberos, on page 1145](#)
- [Monitoring the Kerberos Configuration, on page 1145](#)

Prerequisites for Controlling Switch Access with Kerberos

The following are the prerequisites for controlling switch access with Kerberos.

- So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.
- A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.
- The Kerberos realm name *must* be in all uppercase characters.

Information about Kerberos

This section provides Kerberos information.

Kerberos and Switch Access

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party.



Note In the Kerberos configuration examples, the trusted third party can be any switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

Kerberos Overview

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

Kerberos verifies that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited life span, are stored in user credential caches. The Kerberos server uses the tickets instead of user names and passwords to authenticate users and network services.



Note A Kerberos server can be any switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

Kerberos supports these network services:

- Telnet
- rlogin
- rsh

This table lists the common Kerberos-related terms and definitions.

Table 126: Kerberos Terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch identifies what privileges the user has in a network or on the switch and what actions the user can perform.

Term	Definition
Credential	A general term that refers to authentication tickets, such as TGTs ⁹ and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of re-entering a username and password. Credentials have a default life span of eight hours.
Instance	An authorization level label for Kerberos principals. Most Kerberos principals are of the form <i>user@REALM</i> (for example, <i>smith@EXAMPLE.COM</i>). A Kerberos principal with a Kerberos instance has the form <i>user/instance@REALM</i> (for example, <i>smith/admin@EXAMPLE.COM</i>). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. Note The Kerberos principal and instance names <i>must</i> be in all lowercase characters. Note The Kerberos realm name <i>must</i> be in all uppercase characters.
KDC ¹⁰	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. Note The Kerberos realm name <i>must</i> be in all uppercase characters.
Kerberos server	A daemon that is running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.
KEYTAB ¹¹	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB ¹² .
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. Note The Kerberos principal name <i>must</i> be in all lowercase characters.
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.

Term	Definition
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

⁹ ticket granting ticket

¹⁰ key distribution center

¹¹ key table

¹² server table

Kerberos Operation

A Kerberos server can be a device that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a device as a Kerberos server, remote users must follow these steps:

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. This process then occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.
3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT that includes the user identity to the switch.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
 - If the decryption is successful, the user is authenticated to the switch.
 - If the decryption is not successful, the user repeats Step 2 either by re-entering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different username and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

For instructions about how to authenticate to a KDC, see the “Obtaining a TGT from a KDC” section in the “Security Server Protocols” chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

For instructions about how to authenticate to a network service, see the “Authenticating to Network Services” section in the “Security Server Protocols” chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

How to Configure Kerberos

To set up a Kerberos-authenticated server-client system, follow these steps:

- Configure the KDC by using Kerberos commands.
- Configure the switch to use the Kerberos protocol.

Monitoring the Kerberos Configuration

To display the Kerberos configuration, use the following commands:

- **show running-config**
- **show kerberos creds**: Lists the credentials in a current user’s credentials cache.
- **clear kerberos creds**: Destroys all credentials in a current user’s credentials cache, including those forwarded.



CHAPTER 56

Configuring Local Authentication and Authorization

- [How to Configure Local Authentication and Authorization, on page 1147](#)
- [Monitoring Local Authentication and Authorization, on page 1149](#)

How to Configure Local Authentication and Authorization

Configuring the Switch for Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.



Note To secure the switch for HTTP access by using AAA methods, you must configure the switch with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the switch for HTTP access by using AAA methods.

Follow these steps to configure AAA to operate without a server by setting the switch to implement AAA in local mode:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default local**
5. **aaa authorization exec default local**
6. **aaa authorization network default local**
7. **username name [privilege level] {password encryption-type password}**
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Switch(config)# aaa new-model	Enables AAA.
Step 4	aaa authentication login default local Example: Switch(config)# aaa authentication login default local	Sets the login authentication to use the local username database. The default keyword applies the local user database authentication to all ports.
Step 5	aaa authorization exec default local Example: Switch(config)# aaa authorization exec default local	Configures user AAA authorization, check the local database, and allow the user to run an EXEC shell.
Step 6	aaa authorization network default local Example: Switch(config)# aaa authorization network default local	Configures user AAA authorization for all network-related service requests.
Step 7	username name [privilege level] {password encryption-type password} Example: Switch(config)# username your_user_name privilege 1 password 7 secret567	Enters the local database, and establishes a username-based authentication system. Repeat this command for each user. <ul style="list-style-type: none"> • For <i>name</i>, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For <i>level</i>, specify the privilege level the user has after gaining access. The range is 0 to 15.

	Command or Action	Purpose
		<p>Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access.</p> <ul style="list-style-type: none"> For <i>encryption-type</i>, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. For <i>password</i>, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.
Step 8	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 10	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring Local Authentication and Authorization

To display Local Authentication and Authorization configuration, use the **show running-config** privileged EXEC command.



CHAPTER 57

Configuring Secure Shell

- [Finding Feature Information, on page 1151](#)
- [Prerequisites for Configuring Secure Shell, on page 1151](#)
- [Restrictions for Configuring Secure Shell, on page 1152](#)
- [Information About Configuring Secure Shell , on page 1152](#)
- [How to Configure SSH, on page 1154](#)
- [Monitoring the SSH Configuration and Status, on page 1158](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Configuring Secure Shell

The following are the prerequisites for configuring the switch for secure shell (SSH):

- For SSH to work, the switch needs an Rivest, Shamir, and Adleman (RSA) public/private key pair. This is the same with Secure Copy Protocol (SCP), which relies on SSH for its secure transport.
- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.
- SCP relies on SSH for security.
- SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.
- A user must have appropriate authorization to use SCP.

- A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.
- The Secure Shell (SSH) server requires an IPsec (Data Encryption Standard [DES] or 3DES) encryption software image; the SSH client requires an IPsec (DES or 3DES) encryption software image.)
- Configure a hostname and host domain for your device by using the **hostname** and **ip domain-name** commands in global configuration mode.

Restrictions for Configuring Secure Shell

The following are restrictions for configuring the device for secure shell.

- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on Data Encryption Standard (DES) (56-bit) and 3DES (168-bit) data encryption software. In DES software images, DES is the only encryption algorithm available. In 3DES software images, both DES and 3DES encryption algorithms are available.
- The device supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.
- When using SCP, you cannot enter the password into the **copy** command. You must enter the password when prompted.
- The login banner is not supported in Secure Shell Version 1. It is supported in Secure Shell Version 2.
- The **-l** keyword and **userid** : {number} {ip-address} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for console access.
- To authenticate clients with freeradius over RADSEC, you should generate an RSA key longer than 1024 bit. Use the **crypto key generate rsa general-keys exportable label label-name** command to achieve this.

Information About Configuring Secure Shell

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH and Device Access

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH functions the same in IPv6 as in IPv4. For IPv6, SSH supports IPv6 addresses and enables secure, encrypted connections with remote IPv6 nodes over an IPv6 transport.

SSH Servers, Integrated Clients, and Supported Versions

The Secure Shell (SSH) Integrated Client feature is an application that runs over the SSH protocol to provide device authentication and encryption. The SSH client enables a Cisco device to make a secure, encrypted connection to another Cisco device or to any other device running the SSH server. This connection provides functionality similar to that of an outbound Telnet connection except that the connection is encrypted. With authentication and encryption, the SSH client allows for secure communication over an unsecured network.

The SSH server and SSH integrated client are applications that run on the switch. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client works with publicly and commercially available SSH servers. The SSH client supports the ciphers of Data Encryption Standard (DES), 3DES, and password authentication.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.



Note The SSH client functionality is available only when the SSH server is enabled.

User authentication is performed like that in the Telnet session to the device. SSH also supports the following user authentication methods:

- TACACS+
- RADIUS
- Local authentication and authorization

SSH Configuration Guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If the SSH server is running on an active switch and the active switch fails, the new active switch uses the RSA key pair generated by the previous active switch.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command.
- When generating the RSA key pair, the message No host name specified might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message No domain specified might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

Secure Copy Protocol Overview

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.



Note When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

Secure Copy Protocol

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying device configurations or switch image files. The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the device can determine whether the user has the correct privilege level. To configure the Secure Copy feature, you should understand the SCP concepts.

How to Configure SSH

Setting Up the Switch to Run SSH

Follow the procedure given below to set up your Switch to run SSH:

Before you begin

Configure user authentication for local or remote access. This step is required. For more information, see Related Topics below.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname** *hostname*
4. **ip domain-name** *domain_name*
5. **crypto key generate rsa**

6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>hostname <i>hostname</i></code></p> <p>Example:</p> <pre>Switch(config)# hostname your_hostname</pre>	<p>Configures a hostname and IP domain name for your Switch.</p> <p>Note Follow this procedure only if you are configuring the Switch as an SSH server.</p>
Step 4	<p><code>ip domain-name <i>domain_name</i></code></p> <p>Example:</p> <pre>Switch(config)# ip domain-name your_domain</pre>	<p>Configures a host domain for your Switch.</p>
Step 5	<p><code>crypto key generate rsa</code></p> <p>Example:</p> <pre>Switch(config)# crypto key generate rsa</pre>	<p>Enables the SSH server for local and remote authentication on the Switch and generates an RSA key pair. Generating an RSA key pair for the Switch automatically enables SSH.</p> <p>We recommend that a minimum modulus size of 1024 bits.</p> <p>When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.</p> <p>Note Follow this procedure only if you are configuring the Switch as an SSH server.</p>
Step 6	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
Step 7	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 8	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the SSH Server

Follow the procedure given below to configure the SSH server:



Note This procedure is only required if you are configuring the Switch as an SSH server.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip ssh version [1 | 2]`
4. `ip ssh version [2]`
5. `ip ssh {time-out seconds | authentication-retries number}`
6. Use one or both of the following:
 - `line vty line_number[ending_line_number]`
 - `transport input ssh`
7. `end`
8. `show running-config`
9. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>ip ssh version [1 2]</p> <p>Example:</p> <pre>Switch(config)# ip ssh version 1</pre>	<p>(Optional) Configures the Switch to run SSH Version 1 or SSH Version 2.</p> <ul style="list-style-type: none"> • 1—Configure the Switch to run SSH Version 1. • 2—Configure the Switch to run SSH Version 2. <p>If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.</p>
Step 4	<p>ip ssh version [2]</p> <p>Example:</p> <pre>Switch(config)# ip ssh version 2</pre>	(Optional) Configures the Switch to run SSH Version 2.
Step 5	<p>ip ssh {time-out <i>seconds</i> authentication-retries <i>number</i>}</p> <p>Example:</p> <pre>Switch(config)# ip ssh time-out 90 OR Switch(config)# ip ssh authentication-retries 2</pre>	<p>Configures the SSH control parameters:</p> <ul style="list-style-type: none"> • time-out <i>seconds</i>: Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the Switch uses the default time-out values of the CLI-based sessions. • authentication-retries <i>number</i>: Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5. <p>Repeat this step when configuring both parameters.</p>
Step 6	<p>Use one or both of the following:</p> <ul style="list-style-type: none"> • line vty <i>line_number</i> [<i>ending_line_number</i>] • transport input ssh <p>Example:</p> <pre>Switch(config)# line vty 1 10</pre>	<p>(Optional) Configures the virtual terminal line settings.</p> <ul style="list-style-type: none"> • Enters line configuration mode to configure the virtual terminal line settings. For <i>line_number</i> and <i>ending_line_number</i>, specify a pair of lines. The range is 0 to 15.

	Command or Action	Purpose
	or Switch(config-line) # transport input ssh	<ul style="list-style-type: none"> Specifies that the Switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.
Step 7	end Example: Switch(config-line) # end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Switch# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring the SSH Configuration and Status

This table displays the SSH server configuration and status.

Table 127: Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.



CHAPTER 58

Configuring SSH File Transfer Protocol

Secure Shell (SSH) includes support for SSH File Transfer Protocol (SFTP), which is a new standard file transfer protocol introduced in SSHv2. This feature provides a secure and authenticated method for copying device configuration or device image files.

- [Prerequisites for SSH File Transfer Protocol, on page 1159](#)
- [Restrictions for SSH File Transfer Protocol, on page 1159](#)
- [Information About SSH File Transfer Protocol, on page 1159](#)
- [How to Configure SSH File Transfer Protocol, on page 1160](#)
- [Example: Configuring SSH File Transfer Protocol, on page 1161](#)
- [Additional References, on page 1162](#)
- [Feature Information for SSH File Transfer Protocol, on page 1162](#)

Prerequisites for SSH File Transfer Protocol

- SSH must be enabled.
- The `ip ssh source-interface interface-type interface-number` command must be configured.

Restrictions for SSH File Transfer Protocol

- The SFTP server is not supported.
- SFTP boot is not supported.
- The `sftp` option in the `install add` command is not supported.

Information About SSH File Transfer Protocol

The SFTP client functionality is provided as part of the SSH component and is always enabled on the corresponding device. Therefore, any SFTP server user with the appropriate permission can copy files to and from the device.

An SFTP client is VRF-aware; you can configure the secure FTP client to use the virtual routing and forwarding (VRF) associated with a particular source interface during connection attempts.

How to Configure SSH File Transfer Protocol

The following sections provide information about the various tasks that comprise an SFTP configuration.

Configuring SFTP

Perform the following steps:

Before you begin

To configure a Cisco device for SFTP client-side functionality, the **ip ssh source-interface** *interface-type interface-number* command must be configured first.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ssh source-interface** *interface-type interface-number*
4. **exit**
5. **show running-config**
6. **debug ip sftp**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip ssh source-interface <i>interface-type interface-number</i> Example: Device(config)# ip ssh source-interface GigabitEthernet 1/0/1	Defines the source IP for the SSH session.
Step 4	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show running-config Example: Device# show running-config	(Optional) Displays the SFTP client-side functionality.
Step 6	debug ip sftp Example: Device# debug ip sftp	(Optional) Enables SFTP debugging.

Perform an SFTP Copy Operation

SFTP copy takes the IP or hostname of the corresponding server if Domain Name System (DNS) is configured. To perform SFTP copy operations, use the following commands in privileged EXEC mode:

Command	Purpose
Device# copy ios-file-system:file sftp://user:pwd@server-ip//filepath Or Device# copy ios-file-system: sftp:	Copies a file from the local Cisco IOS file system to the server. Specify the username, password, IP address, and filepath of the server.
Device# copy sftp://user:pwd@server-ip //filepath ios-file-system:file Or Device# copy sftp: ios-file-system:	Copies the file from the server to the local Cisco IOS file system. Specify the username, password, IP address, and filepath of the server.

Example: Configuring SSH File Transfer Protocol

The following example shows how to configure the client-side functionality of SFTP:

```
Device> enable
Device# configure terminal
Device(config)# ip ssh source-interface gigabitethernet 1/0/1
Device(config)# exit
```

Additional References

Related Documents

Related Topic	Document Title
Secure Shell Version 1 and 2 Support	<i>Configuring Secure Shell</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for SSH File Transfer Protocol

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 128: Feature Information for SFTP

Feature Name	Releases	Feature Information
SSH File Transfer Protocol (SFTP)	Cisco IOS Release 15.2(7)E	SSH includes support for SFTP, a new standard file transfer protocol introduced in SSHv2.



CHAPTER 59

X.509v3 Certificates for SSH Authentication

The X.509v3 Certificates for SSH Authentication feature uses public key algorithm (PKI) for server and user authentication, and allows the Secure Shell (SSH) protocol to verify the identity of the owner of a key pair via digital certificates, signed and issued by a Certificate Authority (CA).

This module describes how to configure server and user certificate profiles for a digital certificate.

- [Prerequisites for X.509v3 Certificates for SSH Authentication, on page 1163](#)
- [Restrictions for X.509v3 Certificates for SSH Authentication, on page 1163](#)
- [Information About X.509v3 Certificates for SSH Authentication, on page 1164](#)
- [How to Configure X.509v3 Certificates for SSH Authentication, on page 1165](#)
- [Verifying the Server and User Authentication Using Digital Certificates , on page 1168](#)
- [Configuration Examples for X.509v3 Certificates for SSH Authentication, on page 1172](#)
- [Additional References for X.509v3 Certificates for SSH Authentication, on page 1173](#)
- [Feature Information for X.509v3 Certificates for SSH Authentication, on page 1174](#)

Prerequisites for X.509v3 Certificates for SSH Authentication

The X.509v3 Certificates for SSH Authentication feature replaces the **ip ssh server authenticate user** command with the **ip ssh server algorithm authentication** command. Configure the **default ip ssh server authenticate user** command to remove the **ip ssh server authenticate user** command from the configuration. The IOS secure shell (SSH) server will start using the **ip ssh server algorithm authentication** command.

When you configure the **ip ssh server authenticate user** command, the following message is displayed:



Warning

SSH command accepted; but this CLI will be deprecated soon. Please move to new CLI **ip ssh server algorithm authentication**. Please configure the “**default ip ssh server authenticate user**” to make the CLI ineffective.

Restrictions for X.509v3 Certificates for SSH Authentication

- The X.509v3 Certificates for SSH Authentication feature implementation is applicable only on the Cisco IOS Secure Shell (SSH) server side.

- The Cisco IOS SSH server supports only the x509v3-ssh-rsa algorithm-based certificate for server and user authentication.

Information About X.509v3 Certificates for SSH Authentication

X.509v3 Certificates for SSH Authentication Overview

The Secure Shell (SSH) protocol provides a secure remote access connection to network devices. The communication between the client and server is encrypted.

There are two SSH protocols that use public key cryptography for authentication. The Transport Layer Protocol, uses a digital signature algorithm (called the public key algorithm) to authenticate the server to the client. And the User Authentication Protocol uses a digital signature to authenticate (public key authentication) the client to the server.

The validity of the authentication depends upon the strength of the linkage between the public signing key and the identity of the signer. Digital certificates, such as those in X.509 Version 3 (X.509v3), are used to provide identity management. X.509v3 uses a chain of signatures by a trusted root certification authority and intermediate certificate authorities to bind a public signing key to a specific digital identity. This implementation allows the use of a public key algorithm for server and user authentication, and allows SSH to verify the identity of the owner of a key pair via digital certificates, signed and issued by a Certificate Authority (CA).

Server and User Authentication Using X.509v3

For server authentication, the Secure shell (SSH) server sends its own certificate to the SSH client for verification. This server certificate is associated with the trustpoint configured in the server certificate profile (ssh-server-cert-profile-server configuration mode).

For user authentication, the SSH client sends the user's certificate to the IOS SSH server for verification. The SSH server validates the incoming user certificate using public key infrastructure (PKI) trustpoints configured in the server certificate profile (ssh-server-cert-profile-user configuration mode).

By default, certificate-based authentication is enabled for server and user at the IOS SSH server end.

OCSP Response Stapling

The Online Certificate Status Protocol (OCSP) enables applications to determine the (revocation) state of an identified certificate. This protocol specifies the data that needs to be exchanged between an application checking the status of a certificate and the server providing that status. An OCSP client issues a status request to an OCSP responder and suspends acceptance of the certificate until a response is received. An OCSP response at a minimum consists of a responseStatus field that indicates the processing status of the a request.

For the public key algorithms, the key format consists of a sequence of one or more X.509v3 certificates followed by a sequence of zero or more OCSP responses.

The X.509v3 Certificate for SSH Authentication feature uses OCSP Response Stapling. By using OCSP response stapling, a device obtains the revocation information of its own certificate by contacting the OCSP server and then stapling the result along with its certificates and sending the information to the peer rather than having the peer contact the OCSP responder.

How to Configure X.509v3 Certificates for SSH Authentication

Configuring Digital Certificates for Server Authentication

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] | ssh-rsa [x509v3-ssh-rsa]}**
4. **ip ssh server certificate profile**
5. **server**
6. **trustpoint sign *PKI-trustpoint-name***
7. **ocsp-response include**
8. **end**
9. **line vty line_number [ending_line_number]**
10. **transport input ssh**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]} Example: <pre>Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa</pre>	Defines the order of host key algorithms. Only the configured algorithm is negotiated with the Secure Shell (SSH) client. <p>Note The IOS SSH server must have at least one configured host key algorithm:</p> <ul style="list-style-type: none"> • x509v3-ssh-rsa—certificate-based authentication • ssh-rsa—public key-based authentication
Step 4	ip ssh server certificate profile Example: <pre>Switch(config)# ip ssh server certificate profile</pre>	Configures server and user certificate profiles and enters SSH certificate profile configuration mode.

	Command or Action	Purpose
Step 5	server Example: <pre>Switch(ssh-server-cert-profile)# server</pre>	Configures server certificate profile and enters SSH server certificate profile server configuration mode. <ul style="list-style-type: none"> The server profile is used to send out the certificate of the server to the SSH client during server authentication.
Step 6	trustpoint sign <i>PKI-trustpoint-name</i> Example: <pre>Switch(ssh-server-cert-profile-server)# trustpoint sign trust1</pre>	Attaches the public key infrastructure (PKI) trustpoint to the server certificate profile. <ul style="list-style-type: none"> The SSH server uses the certificate associated with this PKI trustpoint for server authentication.
Step 7	ocsp-response include Example: <pre>Switch(ssh-server-cert-profile-server)# ocsp-response include</pre>	(Optional) Sends the Online Certificate Status Protocol (OCSP) response or OCSP stapling along with the server certificate. <p>Note By default, no OCSP response is sent along with the server certificate.</p>
Step 8	end Example: <pre>Switch(ssh-server-cert-profile-server)# end</pre>	Exits SSH server certificate profile server configuration mode and returns to privileged EXEC mode.
Step 9	line vty line_number [ending_line_number] Example: <pre>Switch(config)# line vty line_number [ending_line_number]</pre>	Enters line configuration mode to configure the virtual terminal line settings. For line_number and ending_line_number, specify a pair of lines. The range is 0 to 15.
Step 10	transport input ssh Example: <pre>Switch(config-line)#transport input ssh</pre>	Specifies that the Switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.

Configuring Digital Certificates for User Authentication

SUMMARY STEPS

- enable
- configure terminal
- ip ssh server algorithm authentication {publickey | keyboard | password}
- ip ssh server algorithm publickey {x509v3-ssh-rsa [ssh-rsa] | ssh-rsa [x509v3-ssh-rsa]}
- ip ssh server certificate profile
- user
- trustpoint verify *PKI-trustpoint-name*
- ocsp-response required

9. `end`
10. `line vty line_number [ending_line_number]`
11. `transport input ssh`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip ssh server algorithm authentication {publickey keyboard password}</p> <p>Example:</p> <pre>Switch(config)# ip ssh server algorithm authentication publickey</pre>	<p>Defines the order of user authentication algorithms. Only the configured algorithm is negotiated with the Secure Shell (SSH) client.</p> <p>Note</p> <ul style="list-style-type: none"> • The IOS SSH server must have at least one configured user authentication algorithm. • To use the certificate method for user authentication, the publickey keyword must be configured.
Step 4	<p>ip ssh server algorithm publickey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]}</p> <p>Example:</p> <pre>Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa</pre>	<p>Defines the order of public key algorithms. Only the configured algorithm is accepted by the SSH client for user authentication.</p> <p>Note</p> <p>The IOS SSH client must have at least one configured public key algorithm:</p> <ul style="list-style-type: none"> • x509v3-ssh-rsa—Certificate-based authentication • ssh-rsa—Public-key-based authentication
Step 5	<p>ip ssh server certificate profile</p> <p>Example:</p> <pre>Switch(config)# ip ssh server certificate profile</pre>	<p>Configures server certificate profile and user certificate profile and enters SSH certificate profile configuration mode.</p>
Step 6	<p>user</p> <p>Example:</p> <pre>Switch(ssh-server-cert-profile)# user</pre>	<p>Configures user certificate profile and enters SSH server certificate profile user configuration mode.</p>

	Command or Action	Purpose
Step 7	trustpoint verify <i>PKI-trustpoint-name</i> Example: <pre>Switch(ssh-server-cert-profile-user)# trustpoint verify trust2</pre>	Configures the public key infrastructure (PKI) trustpoint that is used to verify the incoming user certificate. Note Configure multiple trustpoints by executing the same command multiple times. A maximum of 10 trustpoints can be configured.
Step 8	ocsp-response required Example: <pre>Switch(ssh-server-cert-profile-user)# ocsp-response required</pre>	(Optional) Mandates the presence of the Online Certificate Status Protocol (OCSP) response with the incoming user certificate. Note By default, the user certificate is accepted without an OCSP response.
Step 9	end Example: <pre>Switch(ssh-server-cert-profile-user)# end</pre>	Exits SSH server certificate profile user configuration mode and returns to privileged EXEC mode.
Step 10	line vty line_number [<i>ending_line_number</i>] Example: <pre>Switch(config)# line vty line_number [ending_line_number]</pre>	Enters line configuration mode to configure the virtual terminal line settings. For <i>line_number</i> and <i>ending_line_number</i> , specify a pair of lines. The range is 0 to 15.
Step 11	transport input ssh Example: <pre>Switch(config-line)#transport input ssh</pre>	Specifies that the Switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.

Verifying the Server and User Authentication Using Digital Certificates

SUMMARY STEPS

1. **enable**
2. **show ip ssh**
3. **debug ip ssh detail**
4. **show log**
5. **debug ip packet**
6. **show log**

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode.

- Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 **show ip ssh**

Displays the currently configured authentication methods. To confirm the use of certificate-based authentication, ensure that the x509v3-ssh-rsa algorithm is the configured host key algorithm.

Example:

```
Device# show ip ssh

SSH Enabled - version 1.99
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa
Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
```

Step 3 **debug ip ssh detail**

Turns on debugging messages for SSH details.

Example:

```
Device# debug ip ssh detail

ssh detail messages debugging is on
```

Step 4 **show log**

Shows the debug message log.

Example:

```
Device# show log

Syslog logging: enabled (0 messages dropped, 9 messages rate-limited, 0 flushes, 0 overruns, xml
disabled, filtering disabled)

No Active Message Discriminator.

No Inactive Message Discriminator.

Console logging: disabled
Monitor logging: level debugging, 0 messages logged, xml disabled,
                  filtering disabled
Buffer logging:  level debugging, 233 messages logged, xml disabled,
```

Verifying the Server and User Authentication Using Digital Certificates

```

        filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    File logging: disabled
    Persistent logging: disabled

No active filter modules.

    Trap logging: level informational, 174 message lines logged
    Logging Source-Interface:      VRF Name:

Log Buffer (4096 bytes):
5 IST: SSH2 CLIENT 0: SSH2_MSG_KEXINIT sent
*Sep 6 14:44:08.496 IST: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: kex algo =
diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1
*Sep 6 14:44:08.496 IST: SSH2 0: Server certificate trustpoint not found. Skipping hostkey algo =
x509v3-ssh-rsa
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: hostkey algo = ssh-rsa
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: encryption algo = aes128-ctr,aes192-ctr,aes256-ctr
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: mac algo =
hmac-sha2-256,hmac-sha2-512,hmac-sha1,hmac-sha1-96
*Sep 6 14:44:08.496 IST: SSH2 0: SSH2_MSG_KEXINIT sent
*Sep 6 14:44:08.496 IST: SSH2 0: SSH2_MSG_KEXINIT received
*Sep 6 14:44:08.496 IST: SSH2 0: kex: client->server enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.496 IST: SSH2 0: kex: server->client enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.496 IST: SSH2 0: Using hostkey algo = ssh-rsa
*Sep 6 14:44:08.496 IST: SSH2 0: Using kex_algo = diffie-hellman-group-exchange-sha1
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: SSH2_MSG_KEXINIT received
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: kex: server->client enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: kex: client->server enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: Using hostkey algo = ssh-rsa
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: Using kex_algo = diffie-hellman-group-exchange-sha1
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_REQUEST sent
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: Range sent- 2048 < 2048 < 4096
*Sep 6 14:44:08.497 IST: SSH2 0: SSH2_MSG_KEX_DH_GEX_REQUEST received
*Sep 6 14:44:08.497 IST: SSH2 0: Range sent by client is - 2048 < 2048 < 4096
*Sep 6 14:44:08.497 IST: SSH2 0: Modulus size established : 2048 bits
*Sep 6 14:44:08.510 IST: SSH2 0: expecting SSH2_MSG_KEX_DH_GEX_INIT
*Sep 6 14:44:08.510 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_GROUP received
*Sep 6 14:44:08.510 IST: SSH2 CLIENT 0: Server has chosen 2048 -bit dh keys
*Sep 6 14:44:08.523 IST: SSH2 CLIENT 0: expecting SSH2_MSG_KEX_DH_GEX_REPLY
*Sep 6 14:44:08.524 IST: SSH2 0: SSH2_MSG_KEXDH_INIT received
*Sep 6 14:44:08.555 IST: SSH2: kex_derive_keys complete
*Sep 6 14:44:08.555 IST: SSH2 0: SSH2_MSG_NEWKEYS sent
*Sep 6 14:44:08.555 IST: SSH2 0: waiting for SSH2_MSG_NEWKEYS
*Sep 6 14:44:08.555 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_REPLY received
*Sep 6 14:44:08.555 IST: SSH2 CLIENT 0: Skipping ServerHostKey Validation
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: signature length 271
*Sep 6 14:44:08.571 IST: SSH2: kex_derive_keys complete
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: SSH2_MSG_NEWKEYS sent
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: waiting for SSH2_MSG_NEWKEYS
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: SSH2_MSG_NEWKEYS received
*Sep 6 14:44:08.571 IST: SSH2 0: SSH2_MSG_NEWKEYS received
*Sep 6 14:44:08.571 IST: SSH2 0: Authentications that can continue =
publickey,keyboard-interactive,password
*Sep 6 14:44:08.572 IST: SSH2 0: Using method = none
*Sep 6 14:44:08.572 IST: SSH2 0: Authentications that can continue =
publickey,keyboard-interactive,password
*Sep 6 14:44:08.572 IST: SSH2 0: Using method = keyboard-interactive
*Sep 6 14:44:11.983 IST: SSH2 0: authentication successful for cisco
*Sep 6 14:44:11.984 IST: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: cisco] [Source:
192.168.121.40] [localport: 22] at 14:44:11 IST Thu Sep 6 2018
*Sep 6 14:44:11.984 IST: SSH2 0: channel open request

```

```
*Sep 6 14:44:11.985 IST: SSH2 0: pty-req request
*Sep 6 14:44:11.985 IST: SSH2 0: setting TTY - requested: height 24, width 80; set: height 24, width
80
*Sep 6 14:44:11.985 IST: SSH2 0: shell request
*Sep 6 14:44:11.985 IST: SSH2 0: shell message received
*Sep 6 14:44:11.985 IST: SSH2 0: starting shell for vty
*Sep 6 14:44:22.066 IST: %SYS-6-LOGOUT: User cisco has exited tty session 1(192.168.121.40)
*Sep 6 14:44:22.166 IST: SSH0: Session terminated normally
*Sep 6 14:44:22.167 IST: SSH CLIENT0: Session terminated normally
```

Step 5 debug ip packet

Turns on debugging for IP packet details.

Example:

```
Device# debug ip packet
```

Step 6 show log

Shows the debug message log.

Example:

```
Device# show log
```

```
yslog logging: enabled (0 messages dropped, 9 messages rate-limited, 0 flushes, 0 overruns, xml
disabled, filtering disabled)
```

```
No Active Message Discriminator.
```

```
No Inactive Message Discriminator.
```

```
Console logging: disabled
Monitor logging: level debugging, 0 messages logged, xml disabled,
filtering disabled
Buffer logging: level debugging, 1363 messages logged, xml disabled,
filtering disabled
Exception Logging: size (4096 bytes)
Count and timestamp logging messages: disabled
File logging: disabled
Persistent logging: disabled
```

```
No active filter modules.
```

```
Trap logging: level informational, 176 message lines logged
Logging Source-Interface: VRF Name:
```

```
Log Buffer (4096 bytes):
bleid=0, s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.177 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.177 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
```

```
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
```

Configuration Examples for X.509v3 Certificates for SSH Authentication

Example: Configuring Digital Certificates for Server Authentication

```
Switch> enable
Switch# configure terminal
Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# server
Switch(ssh-server-cert-profile-server)# trustpoint sign trust1
```

```
Switch(ssh-server-cert-profile-server)# exit
```

Example: Configuring Digital Certificate for User Authentication

```
Switch> enable
Switch# configure terminal
Switch(config)# ip ssh server algorithm authentication publickey
Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# user
Switch(ssh-server-cert-profile-user)# trustpoint verify trust2
Switch(ssh-server-cert-profile-user)# end
```

Additional References for X.509v3 Certificates for SSH Authentication

Related Documents

Related Topic	Document Title
PKI configuration	Configuring and Managing a Cisco IOS Certificate Server for PKI Deployment

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for X.509v3 Certificates for SSH Authentication

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 129: Feature Information for X509v3 Certificates for SSH Authentication

Feature Name	Releases	Feature Information
X.509v3 Certificates for SSH Authentication	Cisco IOS 15.2(4)E1	<p>The X.509v3 Certificates for SSH Authentication feature uses the X5.09v3 digital certificates in server and user authentication at the SSH server side.</p> <p>The following commands were introduced or modified: ip ssh server algorithm hostkey, ip ssh server algorithm authentication, and ip ssh server certificate profile.</p> <p>This feature was implemented on the following platforms:</p> <ul style="list-style-type: none"> • Catalyst 2960C, 2960CX, 2960P, 2960X, and 2960XR Series Switches • Catalyst 3560CX and 3560X Series Switches • Catalyst 3750X Series Switches • Catalyst 4500E Sup7-E, Sup7L-E, Sup8-E, and 4500X Series Switches • Catalyst 4900M, 4900F-E Series Switches



CHAPTER 60

Configuring Secure Socket Layer HTTP

- [Finding Feature Information, on page 1175](#)
- [Information about Secure Sockets Layer \(SSL\) HTTP, on page 1175](#)
- [How to Configure Secure HTTP Servers and Clients, on page 1178](#)
- [Monitoring Secure HTTP Server and Client Status, on page 1185](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Information about Secure Sockets Layer (SSL) HTTP

Secure HTTP Servers and Clients Overview

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with `https://` instead of `http://`.



Note SSL evolved into Transport Layer Security (TLS) in 1999, but is still used in this particular context.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which, in turn, responds to the original request.

The primary role of the HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.

Certificate Authority Trustpoints

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as *trustpoints*.

When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a Web browser), in turn, has a public key that allows it to authenticate the certificate.

For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint, when you enable a secure HTTP connection, either a temporary or a persistent self-signed certificate for the secure HTTP server (or client) is automatically generated.

- If the switch is not configured with a hostname and a domain name, a temporary self-signed certificate is generated. If the switch reboots, any temporary self-signed certificate is lost, and a new temporary new self-signed certificate is assigned.
- If the switch has been configured with a host and domain name, a persistent self-signed certificate is generated. This certificate remains active if you reboot the switch or if you disable the secure HTTP server so that it will be there the next time you re-enable a secure HTTP connection.



Note The certificate authorities and trustpoints must be configured on each device individually. Copying them from other devices makes them invalid on the switch.

When a new certificate is enrolled, the new configuration change is not applied to the HTTPS server until the server is restarted. You can restart the server using either the CLI or by physical reboot. On restarting the server, the switch starts using the new certificate.

If a self-signed certificate has been generated, this information is included in the output of the **show running-config** privileged EXEC command. This is a partial sample output from that command displaying a self-signed certificate.

```
Switch# show running-config
Building configuration...

<output truncated>

crypto pki trustpoint TP-self-signed-3080755072
  enrollment selfsigned
  subject-name cn=IOS-Self-Signed-Certificate-3080755072
  revocation-check none
  rsakeypair TP-self-signed-3080755072
!
```



```

!
crypto ca certificate chain TP-self-signed-3080755072
certificate self-signed 01
 3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D

```

<output truncated>

You can remove this self-signed certificate by disabling the secure HTTP server and entering the **no crypto pki trustpoint TP-self-signed-30890755072** global configuration command. If you later re-enable a secure HTTP server, a new self-signed certificate is generated.



Note The values that follow *TP self-signed* depend on the serial number of the device.

You can use an optional command (**ip http secure-client-auth**) to allow the HTTPS server to request an X.509v3 certificate from the client. Authenticating the client provides more security than server authentication by itself.

CipherSuites

A CipherSuite specifies the encryption algorithm and the digest algorithm to use on a SSL connection. When connecting to the HTTPS server, the client Web browser offers a list of supported CipherSuites, and the client and server negotiate the best encryption algorithm to use from those on the list that are supported by both. For example, Netscape Communicator 4.76 supports U.S. security with RSA Public Key Cryptography, MD2, MD5, RC2-CBC, RC4, DES-CBC, and DES-EDE3-CBC.

For the best possible encryption, you should use a client browser that supports 128-bit encryption, such as Microsoft Internet Explorer Version 5.5 (or later) or Netscape Communicator Version 4.76 (or later). The SSL_RSA_WITH_DES_CBC_SHA CipherSuite provides less security than the other CipherSuites, as it does not offer 128-bit encryption.

The more secure and more complex CipherSuites require slightly more processing time. This list defines the CipherSuites supported by the switch and ranks them from fastest to slowest in terms of router processing load (speed):

1. SSL_RSA_WITH_DES_CBC_SHA—RSA key exchange (RSA Public Key Cryptography) with DES-CBC for message encryption and SHA for message digest
2. SSL_RSA_WITH_NULL_SHA key exchange with NULL for message encryption and SHA for message digest (only for SSL 3.0).
3. SSL_RSA_WITH_NULL_MD5 key exchange with NULL for message encryption and MD5 for message digest (only for SSL 3.0).
4. SSL_RSA_WITH_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest
5. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest
6. SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest

7. `SSL_RSA_WITH_AES_128_CBC_SHA`—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).
8. `SSL_RSA_WITH_AES_256_CBC_SHA`—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).
9. `SSL_RSA_WITH_DHE_AES_128_CBC_SHA`—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).
10. `SSL_RSA_WITH_DHE_AES_256_CBC_SHA`—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).



Note The latest versions of Chrome do not support the four original cipher suites, thus disallowing access to both web GUI and guest portals.

RSA (in conjunction with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

Default SSL Configuration

The standard HTTP server is enabled.

SSL is enabled.

No CA trustpoints are configured.

No self-signed certificates are generated.

SSL Configuration Guidelines

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

Before you configure a CA trustpoint, you should ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

In a switch stack, the SSL session terminates at the active switch.

How to Configure Secure HTTP Servers and Clients

Configuring a CA Trustpoint

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

Beginning in privileged EXEC mode, follow these steps to configure a CA Trustpoint:

SUMMARY STEPS

1. **configure terminal**
2. **hostname** *hostname*
3. **ip domain-name** *domain-name*
4. **crypto key generate rsa**
5. **crypto ca trustpoint** *name*
6. **enrollment url** *url*
7. **enrollment http-proxy** *host-name port-number*
8. **crl query** *url*
9. **primary** *name*
10. **exit**
11. **crypto ca authentication** *name*
12. **crypto ca enroll** *name*
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	hostname <i>hostname</i> Example: Switch(config)# hostname your_hostname	Specifies the hostname of the switch (required only if you have not previously configured a hostname). The hostname is required for security keys and certificates.
Step 3	ip domain-name <i>domain-name</i> Example: Switch(config)# ip domain-name your_domain	Specifies the IP domain name of the switch (required only if you have not previously configured an IP domain name). The domain name is required for security keys and certificates.
Step 4	crypto key generate rsa Example: Switch(config)# crypto key generate rsa	(Optional) Generates an RSA key pair. RSA key pairs are required before you can obtain a certificate for the switch. RSA key pairs are generated automatically. You can use this command to regenerate the keys, if needed.
Step 5	crypto ca trustpoint <i>name</i> Example: Switch(config)# crypto ca trustpoint your_trustpoint	Specifies a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.

	Command or Action	Purpose
Step 6	enrollment url <i>url</i> Example: <pre>Switch(ca-trustpoint)# enrollment url http://your_server:80</pre>	Specifies the URL to which the switch should send certificate requests.
Step 7	enrollment http-proxy <i>host-name port-number</i> Example: <pre>Switch(ca-trustpoint)# enrollment http-proxy your_host 49</pre>	(Optional) Configures the switch to obtain certificates from the CA through an HTTP proxy server. <ul style="list-style-type: none"> • For <i>host-name</i>, specify the proxy server used to get the CA. • For <i>port-number</i>, specify the port number used to access the CA.
Step 8	crl query <i>url</i> Example: <pre>Switch(ca-trustpoint)# crl query ldap://your_host:49</pre>	Configures the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
Step 9	primary <i>name</i> Example: <pre>Switch(ca-trustpoint)# primary your_trustpoint</pre>	(Optional) Specifies that the trustpoint should be used as the primary (default) trustpoint for CA requests. <ul style="list-style-type: none"> • For <i>name</i>, specify the trustpoint that you just configured.
Step 10	exit Example: <pre>Switch(ca-trustpoint)# exit</pre>	Exits CA trustpoint configuration mode and return to global configuration mode.
Step 11	crypto ca authentication <i>name</i> Example: <pre>Switch(config)# crypto ca authentication your_trustpoint</pre>	Authenticates the CA by getting the public key of the CA. Use the same name used in Step 5.
Step 12	crypto ca enroll <i>name</i> Example: <pre>Switch(config)# crypto ca enroll your_trustpoint</pre>	Obtains the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.
Step 13	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the Secure HTTP Server

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP server:

Before you begin

If you are using a certificate authority for certification, you should use the previous procedure to configure the CA trustpoint on the switch before enabling the HTTP server. If you have not configured a CA trustpoint, a self-signed certificate is generated the first time that you enable the secure HTTP server. After you have configured the server, you can configure options (path, access list to apply, maximum number of connections, or timeout policy) that apply to both standard and secure HTTP servers.

To verify the secure HTTP connection by using a Web browser, enter `https://URL`, where the URL is the IP address or hostname of the server switch. If you configure a port other than the default port, you must also specify the port number after the URL. For example:



Note AES256_SHA2 is not supported.

```
https://209.165.129.1026
```

or

```
https://host.domain.com:1026
```

The existing **ip http access-class** *access-list-number* command for specifying the access-list(Only IPv4 ACLs) is going to be deprecated. You can still use this command to specify an access list to allow access to the HTTP server. Two new commands have been introduced to enable support for specifying IPv4 and IPv6 ACLs.

These are **ip http access-class ipv4** *access-list-name* | *access-list-number* for specifying IPv4 ACLs and **ip http access-class ipv6** *access-list-name* for specifying IPv6 ACLs. We recommend using the new CLI to avoid receiving warning messages.

Note the following considerations for specifying access-lists:

- If you specify an access-list that does not exist, the configuration takes place but you receive the below warning message:

```
ACL being attached does not exist, please configure it
```

- If you use the **ip http access-class** command for specifying an access-list for the HTTP server, the below warning message appears:

```
This CLI will be deprecated soon, Please use new CLI ip http  
access-class ipv4/ipv6 <access-list-name>| <access-list-number>
```

- If you use **ip http access-class ipv4** *access-list-name* | *access-list-number* or **ip http access-class ipv6** *access-list-name* , and an access-list was already configured using **ip http access-class** , the below warning message appears:

```
Removing ip http access-class <access-list-number>
```

ip http access-class *access-list-number* and **ip http access-class ipv4** *access-list-name* | *access-list-number* share the same functionality. Each command overrides the configuration of the previous command. The

following combinations between the configuration of the two commands explain the effect on the running configuration:

- If **ip http access-class** *access-list-number* is already configured and you try to configure using **ip http access-class ipv4** *access-list-number* command, the configuration of **ip http access-class** *access-list-number* will be removed and the configuration of **ip http access-class ipv4** *access-list-number* will be added to the running configuration.
- If **ip http access-class** *access-list-number* is already configured and you try to configure using **ip http access-class ipv4** *access-list-name* command, the configuration of **ip http access-class** *access-list-number* will be removed and the configuration of **ip http access-class ipv4** *access-list-name* will be added to the running configuration.
- If **ip http access-class ipv4** *access-list-number* is already configured and you try to configure using **ip http access-class** *access-list-name*, the configuration of **ip http access-class ipv4** *access-list-number* will be removed from configuration and the configuration of **ip http access-class** *access-list-name* will be added to the running configuration.
- If **ip http access-class ipv4** *access-list-name* is already configured and you try to configure using **ip http access-class** *access-list-number*, the configuration of **ip http access-class ipv4** *access-list-name* will be removed from the configuration and the configuration of **ip http access-class** *access-list-number* will be added to the running configuration.

SUMMARY STEPS

1. **show ip http server status**
2. **configure terminal**
3. **ip http secure-server**
4. **ip http secure-port** *port-number*
5. **ip http secure-ciphersuite** {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}
6. **ip http secure-client-auth**
7. **ip http secure-trustpoint** *name*
8. **ip http path** *path-name*
9. **ip http access-class** *access-list-number*
10. **ip http access-class** { **ipv4** {*access-list-number* | *access-list-name*} | **ipv6** {*access-list-name*} }
11. **ip http max-connections** *value*
12. **ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	show ip http server status Example: <pre>Switch# show ip http server status</pre>	(Optional) Displays the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output: <pre>HTTP secure server capability: Present</pre> or

	Command or Action	Purpose
		HTTP secure server capability: Not present
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip http secure-server Example: <pre>Switch(config)# ip http secure-server</pre>	Enables the HTTPS server if it has been disabled. The HTTPS server is enabled by default.
Step 4	ip http secure-port <i>port-number</i> Example: <pre>Switch(config)# ip http secure-port 443</pre>	(Optional) Specifies the port number to be used for the HTTPS server. The default port number is 443. Valid options are 443 or any number in the range 1025 to 65535.
Step 5	ip http secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: <pre>Switch(config)# ip http secure-ciphersuite rc4-128-md5</pre>	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particularly CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 6	ip http secure-client-auth Example: <pre>Switch(config)# ip http secure-client-auth</pre>	(Optional) Configures the HTTP server to request an X.509v3 certificate from the client for authentication during the connection process. The default is for the client to request a certificate from the server, but the server does not attempt to authenticate the client.
Step 7	ip http secure-trustpoint <i>name</i> Example: <pre>Switch(config)# ip http secure-trustpoint your_trustpoint</pre>	<p>Specifies the CA trustpoint to use to get an X.509v3 security certificate and to authenticate the client certificate connection.</p> <p>Note Use of this command assumes you have already configured a CA trustpoint according to the previous procedure.</p>
Step 8	ip http path <i>path-name</i> Example: <pre>Switch(config)# ip http path /your_server:80</pre>	(Optional) Sets a base HTTP path for HTML files. The path specifies the location of the HTTP server files on the local system (usually located in system flash memory).

	Command or Action	Purpose
Step 9	ip http access-class <i>access-list-number</i> Example: <pre>Switch(config)# ip http access-class 2</pre>	(Optional) Specifies an access list to use to allow access to the HTTP server.
Step 10	ip http access-class { ipv4 { <i>access-list-number</i> <i>access-list-name</i> } ipv6 { <i>access-list-name</i> } } Example: <pre>Switch(config)# ip http access-class ipv4 4</pre>	(Optional) Specifies an access list to use to allow access to the HTTP server.
Step 11	ip http max-connections <i>value</i> Example: <pre>Switch(config)# ip http max-connections 4</pre>	(Optional) Sets the maximum number of concurrent connections that are allowed to the HTTP server. We recommend that the value be at least 10 and not less. This is required for the UI to function as expected.
Step 12	ip http timeout-policy idle <i>seconds</i> life <i>seconds</i> requests <i>value</i> Example: <pre>Switch(config)# ip http timeout-policy idle 120 life 240 requests 1</pre>	(Optional) Specifies how long a connection to the HTTP server can remain open under the defined circumstances: <ul style="list-style-type: none"> • idle—the maximum time period when no data is received or response data cannot be sent. The range is 1 to 600 seconds. The default is 180 seconds (3 minutes). • life—the maximum time period from the time that the connection is established. The range is 1 to 86400 seconds (24 hours). The default is 180 seconds. • requests—the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
Step 13	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the Secure HTTP Client

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP client:

Before you begin

The standard HTTP client and secure HTTP client are always enabled. A certificate authority is required for secure HTTP client certification. This procedure assumes that you have previously configured a CA trustpoint on the switch. If a CA trustpoint is not configured and the remote HTTPS server requires client authentication, connections to the secure HTTP client fail.

SUMMARY STEPS

1. `configure terminal`
2. `ip http client secure-trustpoint name`
3. `ip http client secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	ip http client secure-trustpoint <i>name</i> Example: <pre>Switch(config)# ip http client secure-trustpoint your_trustpoint</pre>	(Optional) Specifies the CA trustpoint to be used if the remote HTTP server requests client authentication. Using this command assumes that you have already configured a CA trustpoint by using the previous procedure. The command is optional if client authentication is not needed or if a primary trustpoint has been configured.
Step 3	ip http client secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: <pre>Switch(config)# ip http client secure-ciphersuite rc4-128-md5</pre>	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Monitoring Secure HTTP Server and Client Status

To monitor the SSL secure server and client status, use the privileged EXEC commands in the following table.

Table 130: Commands for Displaying the SSL Secure Server and Client Status

Command	Purpose
<code>show ip http client secure status</code>	Shows the HTTP secure client configuration.
<code>show ip http server secure status</code>	Shows the HTTP secure server configuration.
<code>show running-config</code>	Shows the generated self-signed certificate for secure HTTP connections.



CHAPTER 61

Certification Authority Interoperability

This chapter describes how to configure certification authority (CA) interoperability, which is provided in support of the IPsec protocol. CA interoperability permits Cisco IOS devices and CAs to communicate so that your Cisco IOS device can obtain and use digital certificates from the CA. Although IPsec can be implemented in your network without the use of a CA, using a CA provides manageability and scalability for IPsec.

- [Prerequisites For Certification Authority, on page 1187](#)
- [Restrictions for Certification Authority, on page 1187](#)
- [Information About Certification Authority, on page 1187](#)
- [How to Configure Certification Authority, on page 1190](#)
- [Monitoring and Maintaining Certification Authority, on page 1198](#)

Prerequisites For Certification Authority

You need to have a certification authority (CA) available to your network before you configure this interoperability feature. The CA must support the Public Key Infrastructure (PKI) protocol, and the Simple Certificate Enrollment Protocol (SCEP).

Restrictions for Certification Authority

When configuring your CA, the following restrictions apply:

- This feature should be configured only when you also configure both IPsec and Internet Key Exchange (IKE) in your network.
- The Cisco IOS software does not support CA server public keys greater than 2048 bits.

Information About Certification Authority

CA Supported Standards

Without certification authority (CA) interoperability, Cisco IOS devices could not use CAs when deploying IPsec. CAs provide a manageable, scalable solution for IPsec networks.

Cisco supports the following standards with this feature:

- **IPSec**—IPSec is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers. IPSec provides these security services at the IP layer; it uses Internet Key Exchange to handle negotiation of protocols and algorithms based on local policy, and to generate the encryption and authentication keys to be used by IPSec. IPSec can be used to protect one or more data flows between a pair of hosts, between a pair of security gateways, or between a security gateway and a host.
- **Internet Key Exchange (IKE)**—A hybrid protocol that implements Oakley and Skeme key exchanges inside the Internet Security Association Key Management Protocol (ISAKMP) framework. Although IKE can be used with other protocols, its initial implementation is with the IPSec protocol. IKE provides authentication of the IPSec peers, negotiates IPSec keys, and negotiates IPSec security associations.
- **Public-Key Cryptography Standard #7 (PKCS #7)**—A standard from RSA Data Security, Inc., used to encrypt and sign certificate enrollment messages.
- **Public-Key Cryptography Standard #10 (PKCS #10)**—A standard syntax from RSA Data Security, Inc. for certificate requests.
- **RSA Keys**—RSA is the public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adleman. RSA keys come in pairs: one public key and one private key.
- **X.509v3 certificates**—Certificate support that allows the IPSec-protected network to scale by providing the equivalent of a digital ID card to each device. When two devices wish to communicate, they exchange digital certificates to prove their identity (thus removing the need to manually exchange public keys with each peer or to manually specify a shared key at each peer). These certificates are obtained from a CA. X.509 is part of the X.500 standard of the ITU.

Purpose of CAs

Certificate authorities (CAs) are responsible for managing certificate requests and issuing certificates to participating IPSec network devices. These services provide centralized key management for the participating devices.

CAs simplify the administration of IPSec network devices. You can use a CA with a network containing multiple IPSec-compliant devices such as routers.

Digital signatures, enabled by public key cryptography, provide a means of digitally authenticating devices and individual users. In public key cryptography, such as the RSA encryption system, each user has a key pair containing both a public and a private key. The keys act as complements, and anything encrypted with one of the keys can be decrypted with the other. In simple terms, a signature is formed when data is encrypted with a user's private key. The receiver verifies the signature by decrypting the message with the sender's public key. The fact that the message could be decrypted using the sender's public key indicates that the holder of the private key, the sender, must have created the message. This process relies on the receiver's having a copy of the sender's public key and knowing with a high degree of certainty that it really does belong to the sender and not to someone pretending to be the sender.

Digital certificates provide the link. A digital certificate contains information to identify a user or device, such as the name, serial number, company, department, or IP address. It also contains a copy of the entity's public key. The certificate is itself signed by a certification authority (CA), a third party that is explicitly trusted by the receiver to validate identities and to create digital certificates.

In order to validate the signature of the CA, the receiver must first know the CA's public key. Normally this process is handled out-of-band or through an operation done at installation. For instance, most web browsers are configured with the public keys of several CAs by default. The Internet Key Exchange (IKE), an essential component of IPSec, can use digital signatures to scalably authenticate peer devices before setting up security associations.

Without digital signatures, one must manually exchange either public keys or secrets between each pair of devices that use IPsec to protect communications between them. Without certificates, every new device added to the network requires a configuration change on every other device with which it communicates securely. With digital certificates, each device is enrolled with a certification authority. When two devices wish to communicate, they exchange certificates and digitally sign data to authenticate each other. When a new device is added to the network, one simply enrolls that device with a CA, and none of the other devices needs modification. When the new device attempts an IPsec connection, certificates are automatically exchanged and the device can be authenticated.

Implementing IPsec Without CAs

Without a CA, if you want to enable IPsec services (such as encryption) between two Cisco devices, you must first ensure that each device has the key of the other device (such as an RSA public key or a shared key). This requirement means that you must manually perform one of the following operations:

- At each device, enter the RSA public key of the other device.
- At each device, specify a shared key to be used by both device.

In the above illustration, each device uses the key of the other device to authenticate the identity of the other device; this authentication always occurs when IPsec traffic is exchanged between the two devices.

If you have multiple Cisco devices in a mesh topology and wish to exchange IPsec traffic passing among all of those devices, you must first configure shared keys or RSA public keys among all of those devices.

Every time a new device is added to the IPsec network, you must configure keys between the new device and each of the existing devices. (In Figure 34, four additional two-part key configurations would be required to add a single encrypting device to the network.)

Consequently, the more devices there are that require IPsec services, the more involved the key administration becomes. This approach does not scale well for larger, more complex encrypting networks.

Implementing IPsec With CAs

With a CA, you do not have to configure keys between all the encrypting devices. Instead, you individually enroll each participating device with the CA, requesting a certificate for the device. When this has been accomplished, each participating device can dynamically authenticate all the other participating devices. This process is illustrated in the illustration.

To add a new IPsec device to the network, you need only configure that new device to request a certificate from the CA, instead of making multiple key configurations with all the other existing IPsec devices.

Implementing IPsec with Multiple Root CAs

With multiple root CAs, you no longer have to enroll a device with the CA that issued a certificate to a peer. Instead, you configure a device with multiple CAs that it trusts. Thus, a device can use a configured CA (a trusted root) to verify certificates offered by a peer that were not issued by the same CA defined in the identity of the device.

Configuring multiple CAs allows two or more devices enrolled under different domains (different CAs) to verify the identity of each other when using IKE to set up IPsec tunnels.

Through Simple Certificate Enrollment Protocol (SCEP), each device is configured with a CA (the enrollment CA). The CA issues a certificate to the device that is signed with the private key of the CA. To verify the

certificates of peers in the same domain, the device is also configured with the root certificate of the enrollment CA.

To verify the certificate of a peer from a different domain, the root certificate of the enrollment CA in the domain of the peer must be configured securely in the device.

During Internet Key Exchange (IKE) phase one signature verification, the initiator will send the responder a list of its CA certificates. The responder should send the certificate issued by one of the CAs in the list. If the certificate is verified, the device saves the public key contained in the certificate on its public key ring.

With multiple root CAs, VPN users can establish trust in one domain and easily and securely distribute it to other domains. Thus, the required private communication channel between entities authenticated under different domains can occur.

How CA Certificates Are Used by IPsec Devices

When two IPsec devices want to exchange IPsec-protected traffic passing between them, they must first authenticate each other—otherwise, IPsec protection cannot occur. The authentication is done with IKE.

Without a CA, a device authenticates itself to the remote device using either RSA-encrypted nonces or preshared keys. Both methods require that keys must have been previously configured between the two devices.

With a CA, a device authenticates itself to the remote device by sending a certificate to the remote device and performing some public key cryptography. Each device must send its own unique certificate that was issued and validated by the CA. This process works because the certificate of each device encapsulates the public key of the device, each certificate is authenticated by the CA, and all participating devices recognize the CA as an authenticating authority. This scheme is called IKE with an RSA signature.

Your device can continue sending its own certificate for multiple IPsec sessions, and to multiple IPsec peers until the certificate expires. When its certificate expires, the device administrator must obtain a new one from the CA.

CAs can also revoke certificates for devices that will no longer participate in IPsec. Revoked certificates are not recognized as valid by other IPsec devices. Revoked certificates are listed in a certificate revocation list (CRL), which each peer may check before accepting a certificate from another peer.

Registration Authorities

Some CAs have a registration authority (RA) as part of their implementation. An RA is essentially a server that acts as a proxy for the CA so that CA functions can continue when the CA is offline.

Some of the configuration tasks described in this document differ slightly, depending on whether your CA supports an RA.

How to Configure Certification Authority

Managing NVRAM Memory Usage

Certificates and certificate revocation lists (CRLs) are used by your device when a CA is used. Normally certain certificates and all CRLs are stored locally in the NVRAM of the device, and each certificate and CRL uses a moderate amount of memory.

The following certificates are normally stored at your device:

- Certificate of your device
- Certificate of the CA
- Root certificates obtained from CA servers (all root certificates are saved in RAM after the device has been initialized)
- Two registration authority (RA) certificates (only if the CA supports an RA)

CRLs are normally stored at your device according to the following conditions:

- If your CA does not support an RA, only one CRL gets stored in the device.
- If your CA supports an RA, multiple CRLs can be stored in the device.

In some cases, storing these certificates and CRLs locally will not present any difficulty. In other cases, memory might become a problem—particularly if the CA supports an RA and a large number of CRLs have to be stored on the device. If the NVRAM is too small to store root certificates, only the fingerprint of the root certificate is saved.

To save NVRAM space, specify that certificates and CRLs should not be stored locally, but should be retrieved from the CA when needed. This alternative will save NVRAM space but could result in a slight performance impact. To specify that certificates and CRLs should not be stored locally on your device, but should be retrieved when required, enable query mode.

If you do not enable query mode now, you can do it later even if certificates and CRLs have already been stored on the device. In this case, when you enable query mode, the stored certificates and CRLs are deleted from the device after you save the configuration. (If you copy the configuration to a TFTP site prior to enabling query mode, you can save any stored certificates and CRLs at the TFTP site.)

Before disabling query mode, perform the **copy system:running-config nvram:startup-config** command to save all current certificates and CRLs to NVRAM. Otherwise they could be lost during a reboot.

To specify that certificates and CRLs should not be stored locally on your device, but should be retrieved when required, enable query mode by using the following command in global configuration mode:



Note Query mode may affect availability if the CA is down.

SUMMARY STEPS

1. **crypto ca certificate query**

DETAILED STEPS

	Command or Action	Purpose
Step 1	crypto ca certificate query Example: Device(config)# <code>crypto ca certificate query</code>	Enables query mode, which causes certificates and CRLs not to be stored locally.

Configuring the Device Host Name and IP Domain Name

You must configure the host name and IP domain name of a device if this has not already been done. This is required because the device assigns a fully qualified domain name (FQDN) to the keys and certificates used by IPsec, and the FQDN is based on the host name and IP domain name assigned to the device. For example, a certificate named "device20.example.com" is based on a device host name of "device20" and a device IP domain name of "example.com".

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname *name***
4. **ip domain-name *name***
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	hostname <i>name</i> Example: Device(config)# hostname device1	Configures the host name of the device.
Step 4	ip domain-name <i>name</i> Example: Device(config)# ip domain-name domain.com	Configures the IP domain name of the device.
Step 5	end Example: Device(config)# end	Exits global configuration and returns to privileged EXEC mode.

Generating an RSA Key Pair

Rivest, Shamir, and Adelman (RSA) key pairs are used to sign and encrypt IKE key management messages and are required before obtaining a certificate for your device.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **crypto key generate rsa [usage-keys]**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto key generate rsa [usage-keys] Example: Device(config)# crypto key generate rsa usage-keys	Generates an RSA key pair. <ul style="list-style-type: none"> • Use the usage-keys keyword to specify special-usage keys instead of general-purpose keys.
Step 4	end Example: Device(config)# end	Exits global configuration and returns to privileged EXEC mode.

Declaring a Certification Authority

You should declare one certification authority (CA) to be used by the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ca trustpoint *name***
4. **enrollment url *url***
5. **enrollment command**
6. **exit**
7. **crypto pki trustpoint *name***
8. **crl query ldap://*url*[:*port*]**
9. **enrollment {mode ra | retry count *number* | retry period *minutes* | url *url*}**
10. **enrollment {mode ra | retry count *number* | retry period *minutes* | url *url*}**
11. **revocation-check *method1* [*method2 method3*]**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto ca trustpoint <i>name</i> Example: Device(config)# crypto ca trustpoint ka	Declares the certification authority (CA) that your device should use and enters the CA profile enroll configuration mode.
Step 4	enrollment url <i>url</i> Example: Device(ca-profile-enroll)# enrollment url http://entrust:81	Specifies the URL of the CA server to which enrollment requests are sent.
Step 5	enrollment command Example: Device(ca-profile-enroll)# enrollment command	Specifies the HTTP command that is sent to the CA for enrollment.
Step 6	exit Example: Device(ca-profile-enroll)# exit	Exit CA profile enroll configuration mode and returns to global configuration mode.
Step 7	crypto pki trustpoint <i>name</i> Example: Device(config)# crypto pki trustpoint ka	Declares the trustpoint that your device should use and enters Ca-trustpoint configuration mode.
Step 8	crl query ldap://<i>url</i>[:<i>port</i>] Example: Device(ca-trustpoint)# crl query ldap://bar.cisco.com:3899	Queries the certificate revocation list (CRL) to ensure that the certificate of the peer is not revoked.
Step 9	enrollment {<i>mode ra</i> <i>retry count number</i> <i>retry period minutes</i> <i>url url</i>} Example: Device(ca-trustpoint)# enrollment retry period 2	Specifies the enrollment wait period between certificate request retries.
Step 10	enrollment {<i>mode ra</i> <i>retry count number</i> <i>retry period minutes</i> <i>url url</i>} Example: Device(ca-trustpoint)# enrollment retry count 8	Specifies the number of times a device will resend a certificate request when it does not receive a response from the previous request.

	Command or Action	Purpose
Step 11	revocation-check <i>method1</i> [<i>method2 method3</i>] Example: Device(ca-trustpoint)# revocation-check crl ocsp	Checks the revocation status of a certificate.
Step 12	end Example: Device(ca-trustpoint)# end	Exit CA trustpoint configuration mode and returns to privileged EXEC mode.

Configuring a Root CA (Trusted Root)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ca trustpoint** *name*
4. **revocation-check** *method1* [*method2 method3*]
5. **root tftp** *server-hostname filename*
6. **enrollment http-proxy** *hostname port-number*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto ca trustpoint <i>name</i> Example: Device(config)# crypto ca trustpoint ka	Declares the trustpoint that your device should use and enters CA trustpoint configuration mode.
Step 4	revocation-check <i>method1</i> [<i>method2 method3</i>] Example: Device(ca-trustpoint)# revocation-check ocsp	Checks the revocation status of a certificate.
Step 5	root tftp <i>server-hostname filename</i> Example: Device(ca-trustpoint)# root tftp server1 file1	Obtains the certification authority (CA) certificate via TFTP.

	Command or Action	Purpose
Step 6	enrollment http-proxy <i>hostname port-number</i> Example: Device(ca-trustpoint)# enrollment http-proxy host2 8080	Accesses the certification authority (CA) by HTTP through the proxy server.
Step 7	end Example: Device(ca-trustpoint)# end	Exits CA trustpoint configuration mode and returns to privileged EXEC mode.

Authenticating the CA

The device must authenticate the certification authority (CA). It does this by obtaining the self-signed certificate of the CA, which contains the public key of the CA. Because the certificate of the CA is self-signed (the CA signs its own certificate) the public key of the CA should be manually authenticated by contacting the CA administrator to compare the fingerprint of the CA certificate when you perform this step.

Perform the following task to get the public key of the CA:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki authenticatename**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki authenticatename Example: Device(config)# crypto pki authenticate myca	Authenticates the CA by getting the certificate of the CA.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Requesting Signed Certificates

You must obtain a signed certificate from the certification authority (CA) for each of the RSA key pairs on your device. If you generated general-purpose RSA keys, your device has only one RSA key pair and needs only one certificate. If you previously generated special-usage RSA keys, your device has two RSA key pairs and needs two certificates.

Perform the following task to request signed certificates from the CA:



Note If your device reboots after you have issued the **crypto pki enroll** command, but before you have received the certificates, you must reissue the command and notify the CA administrator.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki enroll** *number*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki enroll <i>number</i> Example: Device(config)# crypto pki enroll myca	Obtains certificates for your device from the CA.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

What to do next

Saving Your Configuration

Always remember to save your work when you make configuration changes.

Use the **copy system:running-config nvram:startup-config** command to save your configuration. This command includes saving RSA keys to private NVRAM. RSA keys are not saved with your configuration when you use a **copy system:running-config rpc:** or **copy system:running-config tftp:** command.

Monitoring and Maintaining Certification Authority

Requesting a Certificate Revocation List

You can request a certificate revocation list (CRL) only if the certification authority (CA) does not support a registration authority (RA). The following task applies only when the CA does not support an RA.

When a device receives a certificate from a peer, your device will download a CRL from the CA. The device then checks the CRL to make sure the certificate that the peer sent has not been revoked. (If the certificate appears on the CRL, the device will not accept the certificate and will not authenticate the peer.)

A CRL can be reused with subsequent certificates until the CRL expires if query mode is off. If the device receives a peer's certificate after the applicable CRL has expired, the device will download the new CRL.

If the device has a CRL that has not yet expired, but you suspect that the contents of the CRL are out of date, you can request that the latest CRL be downloaded immediately to replace the old CRL.

•

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki crl request *name***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki crl request <i>name</i> Example: Device(config)# crypto pki crl request myca	Requests that a new certificate revocation list (CRL) be obtained immediately from the CA.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Querying a Certification Revocation List

You can query a certificate revocation list (CRL) only when you configure your device with a trusted root. When your device receives a certificate from a peer from another domain (with a different CA), the CRL downloaded from the CA of the device will not include certificate information about the peer. Therefore, you should check the CRL published by the configured root with the LDAP URL to ensure that the certificate of the peer has not been revoked.

If you would like CRL of the root certificate to be queried when the device reboots, you must enter the **cr1 query** command.

Perform the following task to query the CRL published by the configured root with the LDAP URL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint *name***
4. **cr1 query ldap *://url* : [*port*]**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>name</i> Example: Device(ca-trustpoint)# crypto pki trustpoint mytp	Declares the trustpoint that your device should use and enters CA trustpoint configuration mode.
Step 4	cr1 query ldap <i>://url</i> : [<i>port</i>] Example: Device(ca-trustpoint)# cr1 query ldap://url:[port]	Queries the CRL to ensure that the certificate of the peer has not been revoked.
Step 5	end Example: Device(ca-trustpoint)# end	Exits CA trustpoint configuration mode and returns to privileged EXEC mode.

Deleting RSA Keys from a Device

Under certain circumstances you may want to delete RSA keys from your device. For example, if you believe the RSA keys were compromised in some way and should no longer be used, you should delete the keys.

]

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto key zeroize rsa** [*key-pair-label*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto key zeroize rsa [<i>key-pair-label</i>] Example: Device(config)# crypto key zeroize rsa	Deletes all Rivest, Shamir, and Adelman (RSA) keys from your device.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

What to do next

After you delete RSA keys from the device, you should also complete the following two additional tasks:

- Ask the CA administrator to revoke the device certificates at the CA; you must supply the challenge password that you created when you originally obtained the device certificates with the **crypto pki enroll** command.
- Manually remove the device certificates from the device configuration.

Deleting Public Keys for a Peer

Under certain circumstances you may want to delete RSA public keys of peer devices from your device configuration. For example, if you no longer trust the integrity of the public key of a peer, you should delete the key.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto key pubkey-chain rsa**

4. **no named key** *key-name* [encryption | signature]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto key pubkey-chain rsa Example: Device(config)# crypto key pubkey-chain rsa	Enters public key chain configuration mode, so that you can manually specify other devices' RSA public keys.
Step 4	no named key <i>key-name</i> [encryption signature] Example: Device(config-pubkey-c)# no named-key otherpeer.example.com	Deletes the RSA public key of a remote peer and enters public key configuration mode.
Step 5	end Example: Device(config-pubkey)# end	Exits public key configuration mode and returns to privileged EXEC mode.

Deleting Certificates from the Configuration

If the need arises, you can delete certificates that are saved in your device. Your devices saves its own certificates, the certificate of the CA, and any RA certificates .

To delete the CA's certificate, you must remove the entire CA identity, which also removes all certificates associated with the CA—your router's certificate, the CA certificate, and any RA certificates.

SUMMARY STEPS

1. **enable**
2. **show crypto pki certificates**
3. **configure terminal**
4. **crypto pki certificate chain** *name*
5. **no certificate** *certificate-serial-number*
6. **exit**
7. **no crypto pki import** *name* certificate
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show crypto pki certificates Example: Device# show crypto pki certificates	Displays information about your device certificate, the certification authority (CA) certificate, and any registration authority (RA) certificates.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	crypto pki certificate chain <i>name</i> Example: Device(config)# crypto pki certificate chain myca	Enters certificate chain configuration mode.
Step 5	no certificate <i>certificate-serial-number</i> Example: Device(config-cert-chain)# no certificate 0123456789ABCDEF0123456789ABCDEF	Deletes the certificate.
Step 6	exit Example: Device(config-cert-chain)# exit	Exits certificate chain configuration mode and returns to global configuration mode.
Step 7	no crypto pki import <i>name</i> certificate Example: Device(config)# no crypto pki import MS certificate	Deletes a certificate manually.
Step 8	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Viewing Keys and Certificates

Perform the following task to view keys and certificates:

SUMMARY STEPS

1. enable
2. show crypto key mypubkey rsa [*keyname*]
3. show crypto key pubkey-chain rsa
4. show crypto key pubkey-chain rsa [*name key-name* | *address key-address*]

5. `show crypto pki certificates`
6. `show crypto pki trustpoints`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show crypto key mypubkey rsa [keyname] Example: Device# show crypto key mypubkey rsa [keyname]	Displays the RSA public keys configured on a device.
Step 3	show crypto key pubkey-chain rsa Example: Device# show crypto key pubkey-chain rsa	Displays the RSA public keys of the peer that are stored on a device.
Step 4	show crypto key pubkey-chain rsa [name key-name address key-address] Example: Device# show crypto key pubkey-chain rsa address 209.165.202.129	Displays the address of a specific key.
Step 5	show crypto pki certificates Example: Device# show crypto pki certificates	Displays information about the device certificate, the certification authority (CA) certificate, and any registration authority (RA) certificates
Step 6	show crypto pki trustpoints Example: Device# show crypto pki certificates	Displays trustpoints that are configured on a device.



CHAPTER 62

Access Control List Overview

Access lists filter network traffic by controlling the forwarding or blocking of packets at the interface of a device. A device examines each packet to determine whether to forward or drop that packet, based on the criteria specified in access lists.

The criteria that can be specified in an access list include the source address of the traffic, the destination address of the traffic, and the upper-layer protocol.



Note Some users might successfully evade basic access lists because these lists require no authentication.

- [Information About Access Control Lists, on page 1205](#)

Information About Access Control Lists

Definition of an Access List

An access list is a sequential list consisting of at least one **permit** statement and possibly one or more **deny** statements. In the case of IP access lists, the statements can apply to IP addresses, upper-layer IP protocols, or other fields in IP packets. The access list is identified and referenced by a name or a number. Access list acts as a packet filter, filtering packets based on the criteria defined in the access list.

An access list may be configured, but it does not take effect until the access list is either applied to an interface, a virtual terminal line (vty), or referenced by some command that accepts an access list. Multiple commands can reference the same access list.

The following configuration example shows how to create an IP access list named `branchoffices`. The ACL is applied to serial interface 0 on incoming packets. No sources other than those on the networks specified by each source address and mask pair can access this interface. The destinations for packets coming from sources on network 172.20.7.0 are unrestricted. The destination for packets coming from sources on network 172.29.2.0 must be 172.25.5.4.

```
ip access-list extended branchoffices
 10 permit 172.20.7.0 0.0.0.3 any
 20 permit 172.29.2.0 0.0.0.255 host 172.25.5.4
!
interface serial 0
 ip access-group branchoffices in
```

Functions of an Access Control List

There are many reasons to configure access lists; for example, to restrict contents of routing updates or to provide traffic flow control. One of the most important reasons to configure access lists is to provide security for your network, which is the focus of this module.

Use access lists to provide a basic level of security for accessing your network. If you do not configure access lists on your device, all packets passing through the device are allowed access to all parts of your network.

Access lists can allow a host to access a part of your network and prevent another host from accessing the same area. In the figure below, Host A is allowed to access the Human Resources network, but Host B is prevented from accessing the Human Resources network.

You can also use access lists to define the type of traffic that is forwarded or blocked at device interfaces. For example, you can permit e-mail traffic to be routed but at the same time block all Telnet traffic.

Purpose of IP Access Lists

Access lists perform packet filtering to control which packets move through the network and where. Such control can help limit network traffic and restrict the access of users and devices to the network. Access lists have many uses, and therefore many commands accept a reference to an access list in their command syntax. Access lists can be used to do the following:

- Filter incoming packets on an interface.
- Filter outgoing packets on an interface.
- Restrict the contents of routing updates.
- Limit debug output based on an address or protocol.
- Control virtual terminal line access.
- Identify or classify traffic for advanced features, such as congestion avoidance, congestion management, and priority and custom queuing.
- Trigger dial-on-demand routing (DDR) calls.

Reasons to Configure ACLs

There are many reasons to configure access lists; for example, you can use access lists to restrict contents of switching updates or to provide traffic flow control. One of the most important reasons to configure access lists is to provide a basic level of security for your network by controlling access to it. If you do not configure access lists on your device, all packets passing through the device could be allowed onto all parts of your network.

An access list can allow one host to access a part of your network and prevent another host from accessing the same area. For example, by applying an appropriate access list to interfaces of a device, Host A is allowed to access the human resources network and Host B is prevented from accessing the human resources network.

You can use access lists on a device that is positioned between two parts of your network, to control traffic entering or exiting a specific part of your internal network.

To provide some security benefits of access lists, you should at least configure access lists on border devices—devices located at the edges of your networks. Such an access list provides a basic buffer from the

outside network or from a less controlled area of your own network into a more sensitive area of your network. On these border devices, you should configure access lists for each network protocol configured on the device interfaces. You can configure access lists so that inbound traffic or outbound traffic or both are filtered on an interface.

Access lists are defined on a per-protocol basis. In other words, you should define access lists for every protocol enabled on an interface if you want to control traffic flow for that protocol.

Software Processing of an Access List

The following general steps describe how the an access list is processed when it is applied to an interface, a vty, or referenced by any command. These steps apply to an access list that has 13 or fewer access list entries.

- The software receives an IP packet and tests parts of each packet being filtered against the conditions in the access list, one condition (**permit** or **deny** statement) at a time. For example, the software tests the source and destination addresses of the packet against the source and destination addresses in a **permit** or **deny** statement.
- If a packet does not match an access list statement, the packet is then tested against the next statement in the list.
- If a packet and an access list statement match, the rest of the statements in the list are skipped and the packet is permitted or denied as specified in the matched statement. The first entry that the packet matches determines whether the software permits or denies the packet. That is, after the first match, no subsequent entries are considered.
- If the access list denies a packet, the software discards the packet and returns an Internet Control Message Protocol (ICMP) Host Unreachable message.
- If no conditions match, the software drops the packet. This is because each access list ends with an unwritten, implicit **deny** statement. That is, if the packet has not been permitted by the time it was tested against each statement, it is denied.

An access list with more than 13 entries is processed using a trie-based lookup algorithm. This process will happen automatically; it does not need to be configured.

Access List Rules

The following rules apply to access control lists (ACLs):

- Only one access list per interface, per protocol, and per direction is allowed.
- An access list must contain at least one **permit** statement or all packets are denied entry into the network.
- The order in which access list conditions or match criteria are configured is important. While deciding whether to forward or block a packet, Cisco software tests the packet against each criteria statement in the order in which these statements are created. After a match is found, no more criteria statements are checked. The same **permit** or **deny** statements specified in a different order can result in a packet being passed under one circumstance and denied in another circumstance.
- If an access list is referenced by a name, but the access list does not exist, all packets pass. An interface or command with an empty access list applied to it permits all traffic into the network.
- Standard access lists and extended access lists cannot have the same name.

- Inbound access lists process packets before packets are sent to an outbound interface. Inbound access lists that have filtering criteria that deny packet access to a network saves the overhead of a route lookup. Packets that are permitted access to a network based on the configured filtering criteria are processed for routing. For inbound access lists, when you configure a **permit** statement, packets are processed after they are received, and when you configure a **deny** statement, packets are discarded.
- Outbound access lists process packets before they leave the device. Incoming packets are routed to the outbound interface and then processed by the outbound access list. For outbound access lists, when you configure a **permit** statement, packets are sent to the output buffer, and when you configure a **deny** statement, packets are discarded.
- An access list can control traffic arriving at a device or leaving a device, but not traffic originating at a device.

Helpful Hints for Creating IP Access Lists

The following tips will help you avoid unintended consequences and help you create more efficient access lists.

- Create the access list before applying it to an interface (or elsewhere), because if you apply a nonexistent access list to an interface and then proceed to configure the access list, the first statement is put into effect, and the implicit **deny** statement that follows could cause you immediate access problems.
- Another reason to configure an access list before applying it is because an interface with an empty access list applied to it permits all traffic.
- All access lists need at least one **permit** statement; otherwise, all packets are denied and no traffic passes.
- Because the software stops testing conditions after it encounters the first match (to either a **permit** or **deny** statement), you will reduce processing time and resources if you put the statements that packets are most likely to match at the beginning of the access list. Place more frequently occurring conditions before less frequent conditions.
- Organize your access list so that more specific references in a network or subnet appear before more general ones.
- Use the statement **permit any any** if you want to allow all other packets not already denied. Using the statement **permit any any** in effect avoids denying all other packets with the implicit deny statement at the end of an access list. Do not make your first access list entry **permit any any** because all traffic will get through; no packets will reach the subsequent testing. In fact, once you specify **permit any any**, all traffic not already denied will get through.
- Although all access lists end with an implicit **deny** statement, we recommend use of an explicit **deny** statement (for example, **deny ip any any**). On most platforms, you can display the count of packets denied by issuing the **show access-list** command, thus finding out more information about who your access list is disallowing. Only packets denied by explicit **deny** statements are counted, which is why the explicit **deny** statement will yield more complete data for you.
- While you are creating an access list or after it is created, you might want to delete an entry.
 - You cannot delete an entry from a numbered access list; trying to do so will delete the entire access list. If you need to delete an entry, you need to delete the entire access list and start over.
 - You can delete an entry from a named access list. Use the **no permit** or **no deny** command to delete the appropriate entry.

- In order to make the purpose of individual statements more scannable and easily understood at a glance, you can write a helpful remark before or after any statement by using the **remark** command.
- If you want to deny access to a particular host or network and find out if someone from that network or host is attempting to gain access, include the **log** keyword with the corresponding **deny** statement so that the packets denied from that source are logged for you.
- This hint applies to the placement of your access list. When trying to save resources, remember that an inbound access list applies the filter conditions before the routing table lookup. An outbound access list applies the filter conditions after the routing table lookup.

IP Packet Fields You Can Filter to Control Access

You can use an extended access list to filter on any of the following fields in an IP packet. Source address and destination address are the two most frequently specified fields on which to base an access list:

- Source address--Specifies a source address to control packets coming from certain networking devices or hosts.
- Destination address--Specifies a destination address to control packets being sent to certain networking devices or hosts.
- Protocol--Specifies an IP protocol indicated by the keyword **eigrp**, **gre**, **icmp**, **igmp**, **ip**, **ipinip**, **nos**, **ospf**, **tcp**, or **udp**, or indicated by an integer in the range from 0 to 255 (representing an Internet protocol). If you specify a transport layer protocol (**icmp**, **igmp**, **tcp**, or **udp**), the command has a specific syntax.
 - Ports and non-contiguous ports--Specifies TCP or UDP ports by a port name or port number. The port numbers can be noncontiguous port numbers. Port numbers can be useful to filter Telnet traffic or HTTP traffic, for example.
 - TCP flags--Specifies that packets match any flag or all flags set in TCP packets. Filtering on specific TCP flags can help prevent false synchronization packets.
- IP options--Specifies IP options; one reason to filter on IP options is to prevent routers from being saturated with spurious packets containing them.

Source and Destination Addresses

Source and destination address fields in an IP packet are two typical fields on which to base an access list. Specify source addresses to control the packets being sent from certain networking devices or hosts. Specify destination addresses to control the packets being sent to certain networking devices or hosts.

Wildcard Mask for Addresses in an Access List

Address filtering uses wildcard masking to indicate to the software whether to check or ignore corresponding IP address bits when comparing the address bits in an access list entry to a packet being submitted to the access list. By carefully setting wildcard masks, you can specify one or more IP addresses for permit or deny tests.

Wildcard masking for IP address bits uses the number 1 and the number 0 to specify how the software treats the corresponding IP address bits. A wildcard mask is sometimes referred to as an inverted mask because a 1 and 0 mean the opposite of what they mean in a subnet (network) mask.

- A wildcard mask bit 0 means check the corresponding bit value; they must match.
- A wildcard mask bit 1 means ignore that corresponding bit value; they need not match.

If you do not supply a wildcard mask with a source or destination address in an access list statement, the software assumes an implicit wildcard mask of 0.0.0.0, meaning all values must match.

Unlike subnet masks, which require contiguous bits indicating network and subnet to be ones, wildcard masks allow noncontiguous bits in the mask.

The table below shows examples of IP addresses and masks from an access list, along with the corresponding addresses that are considered a match.

Table 131: Sample IP Addresses, Wildcard Masks, and Match Results

Address	Wildcard Mask	Match Results
0.0.0.0	255.255.255.255	All addresses will match the access list conditions.
172.18.0.0/16	0.0.255.255	Network 172.18.0.0
172.18.5.2/16	0.0.0.0	Only host 172.18.5.2 matches
172.18.8.0	0.0.0.7	Only subnet 172.18.8.0/29 matches
172.18.8.8	0.0.0.7	Only subnet 172.18.8.8/29 matches
172.18.8.15	0.0.0.3	Only subnet 172.18.8.15/30 matches
10.1.2.0	0.0.254.255 (noncontiguous bits in mask)	Matches any even-numbered network in the range of 10.1.2.0 to 10.1.254.0

Access List Sequence Numbers

The ability to apply sequence numbers to IP access list entries simplifies access list changes. Prior to the IP Access List Entry Sequence Numbering feature, there was no way to specify the position of an entry within an access list. If you wanted to insert an entry in the middle of an existing list, all of the entries after the desired position had to be removed, then the new entry was added, and then all the removed entries had to be reentered. This method was cumbersome and error prone.

This feature allows users to add sequence numbers to access list entries and resequence them. When you add a new entry, you specify the sequence number so that it is in a desired position in the access list. If necessary, entries currently in the access list can be resequenced to create room to insert the new entry.

ACL Supported Types

The switch supports IP ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs.

Supported ACLs

The switch supports three types of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. You can apply port ACLs to a Layer 2 interface in each direction to each access list type — IPv4 and MAC.
- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound).

ACL Precedence

When VLAN maps, Port ACLs, and router ACLs are configured on the same switch, the filtering precedence, from greatest to least for ingress traffic is port ACL, VLAN map, and then router ACL. For egress traffic, the filtering precedence is router ACL, VLAN map, and then port ACL.

The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

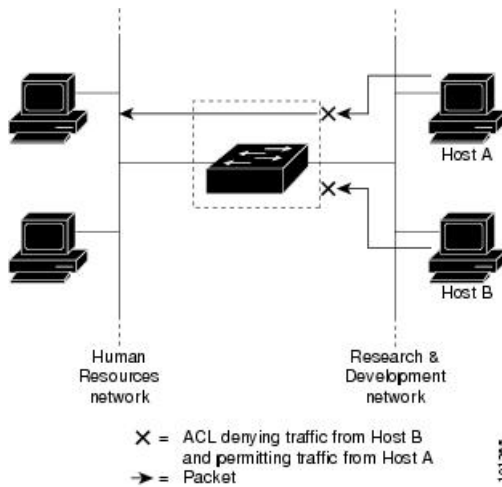
Port ACLs

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces. Port ACLs can be applied to the interface in inbound direction. The following access lists are supported:

- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

The switch examines ACLs on an interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs control access to a network or to part of a network.

Figure 93: Using ACLs to Control Traffic in a Network



This is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the inbound direction.

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.



Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

You can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL, and can be used to control access to a network or to part of a network.

Access Control Entries

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

ACEs and Fragmented and Unfragmented Traffic

IP packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some access control entries (ACEs) do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IP packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.



Note For TCP ACEs with L4 Ops, the fragmented packets will be dropped per RFC 1858.

- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

ACEs and Fragmented and Unfragmented Traffic Examples

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Switch(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Switch(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Switch(config)# access-list 102 permit tcp any host 10.1.1.2
Switch(config)# access-list 102 deny tcp any any
```



Note In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2., port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer

3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.

- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.

- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.



CHAPTER 63

IPv4 ACLs

- [Finding Feature Information, on page 1215](#)
- [Restrictions for Configuring IPv4 Access Control Lists, on page 1215](#)
- [Information about Network Security with ACLs, on page 1216](#)
- [How to Configure ACLs, on page 1228](#)
- [Monitoring IPv4 ACLs, on page 1250](#)
- [Configuration Examples for ACLs, on page 1251](#)
- [Feature Information for IPv4 Access Control Lists, on page 1265](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restrictions for Configuring IPv4 Access Control Lists

General Network Security

The following are restrictions for configuring network security with ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name.
- A standard ACL and an extended ACL cannot have the same name.
- Though visible in the command-line help strings, **AppleTalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.
- ACL wild card is not supported in downstream client policy.

IPv4 ACL Network Interfaces

The following restrictions apply to IPv4 ACLs to network interfaces:

- When controlling access to an interface, you can use a named or numbered ACL.
- If you apply an ACL to a Layer 3 interface and routing is not enabled on the switch, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or web traffic.
- You do not have to enable routing to apply ACLs to Layer 2 interfaces.

MAC ACLs on a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface. When you apply the MAC ACL, consider these guidelines:

- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.



Note The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface. You cannot use the command on EtherChannel port channels.

IP Access List Entry Sequence Numbering

- This feature does not support dynamic, reflexive, or firewall access lists.

Information about Network Security with ACLs

This chapter describes how to configure network security on the switch by using access control lists (ACLs), which in commands and tables are also referred to as access lists.

Cisco TrustSec and ACLs

Catalyst 3850 switches running the IP base or IP services feature set also support Cisco TrustSec Security Group Tag (SCT) Exchange Protocol (SXP). This feature supports security group access control lists (SGACLs), which define ACL policies for a group of devices instead of an IP address. The SXP control protocol allows tagging packets with SCTs without a hardware upgrade, and runs between access layer devices at the Cisco TrustSec domain edge and distribution layer devices within the Cisco TrustSec domain. Catalyst 3850 switches operate as access layer switches in the Cisco TrustSec network.

The sections on SXP define the capabilities supported on the Catalyst 3850 switches.

ACL Overview

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic.

Access Control Entries

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

ACL Supported Types

The switch supports IP ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs.

Supported ACLs

The switch supports three types of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. You can apply port ACLs to a Layer 2 interface in each direction to each access list type — IPv4 and MAC.
- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound).

ACL Precedence

When VLAN maps, Port ACLs, and router ACLs are configured on the same switch, the filtering precedence, from greatest to least for ingress traffic is port ACL, VLAN map, and then router ACL. For egress traffic, the filtering precedence is router ACL, VLAN map, and then port ACL.

The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

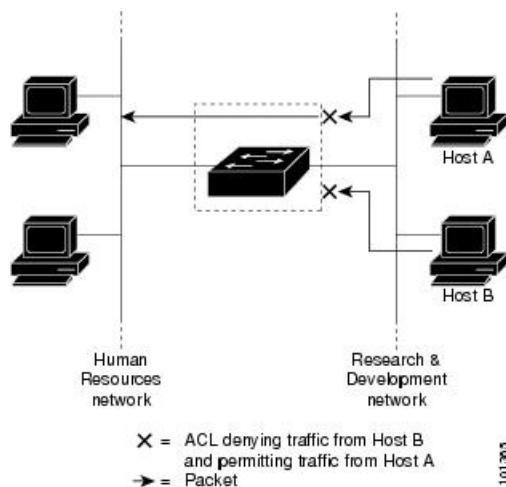
Port ACLs

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces. Port ACLs can be applied to the interface in inbound direction. The following access lists are supported:

- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

The switch examines ACLs on an interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs control access to a network or to part of a network.

Figure 94: Using ACLs to Control Traffic in a Network



This is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the inbound direction.

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.



Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

You can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL, and can be used to control access to a network or to part of a network.

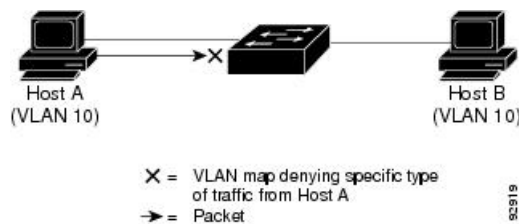
VLAN Maps

VLAN ACLs or VLAN maps are used to control network traffic within a VLAN. You can apply VLAN maps to all packets that are bridged within a VLAN in the switch or switch stack. VACLs are strictly for security packet filtering and for redirecting traffic to specific physical interfaces. VACLs are not defined by direction (ingress or egress).

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic is not access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map.

Figure 95: Using VLAN Maps to Control Traffic



This figure shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.

ACEs and Fragmented and Unfragmented Traffic

IP packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some access control entries (ACEs) do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IP packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.



Note For TCP ACEs with L4 Ops, the fragmented packets will be dropped per RFC 1858.

- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

ACEs and Fragmented and Unfragmented Traffic Examples

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Switch(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Switch(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Switch(config)# access-list 102 permit tcp any host 10.1.1.2
Switch(config)# access-list 102 deny tcp any any
```



Note In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2, port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer 3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.
- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.
- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.

Standard and Extended IPv4 ACLs

This section describes IP ACLs.

An ACL is a sequential collection of permit and deny conditions. One by one, the switch tests packets against the conditions in an access list. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports these types of ACLs or access lists for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

IPv4 ACL Switch Unsupported Features

Configuring IPv4 ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers.

The following ACL-related features are not supported:

- Non-IP protocol ACLs
- IP accounting
- Reflexive ACLs, URL Redirect ACLs and Dynamic ACLs are not supported.
- ACL logging for port ACLs and VLAN maps

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating.

This lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 132: Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

In addition to numbered standard and extended ACLs, you can also create standard and extended named IP ACLs by using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Numbered Standard IPv4 ACLs

When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

After creating a numbered standard IPv4 ACL, you can apply it to VLANs, to terminal lines, or to interfaces.

Numbered Extended IPv4 ACLs

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Some protocols also have specific parameters and keywords that apply to that protocol.

You can define an extended TCP, UDP, ICMP, IGMP, or other IP ACL. The switch also supports these IP protocols:



Note ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

These IP protocols are supported:

- Authentication Header Protocol (**ahp**)
- Encapsulation Security Payload (**esp**)
- Enhanced Interior Gateway Routing Protocol (**eigrp**)
- generic routing encapsulation (**gre**)
- Internet Control Message Protocol (**icmp**)
- Internet Group Management Protocol (**igmp**)
- any Interior Protocol (**ip**)
- IP in IP tunneling (**ipinip**)
- KA9Q NOS-compatible IP over IP tunneling (**nos**)

- Open Shortest Path First routing (**ospf**)
- Payload Compression Protocol (**pcp**)
- Protocol-Independent Multicast (**pim**)
- Transmission Control Protocol (**tcp**)
- User Datagram Protocol (**udp**)

Named IPv4 ACLs

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.



Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99 and . The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines before configuring named ACLs:

- Numbered ACLs are also available.
- A standard ACL and an extended ACL cannot have the same name.
- You can use standard or extended ACLs (named or numbered) in VLAN maps.

ACL Logging

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the **logging console** commands controlling the syslog messages.



Note ACL logging is only supported for RACL.



Note Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message right away, and subsequent packets are collected over 5-minute intervals before they appear or logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.



Note The logging facility might drop some logging message packets if there are too many to be handled or if there is more than one logging message to be handled in 1 second. This behavior prevents the router from crashing due to too many logging packets. Therefore, the logging facility should not be used as a billing tool or an accurate source of the number of matches to an access list.

Smart Logging

When smart logging is enabled on the switch and an ACL configured with smart logging is attached to a Layer 2 interface (port ACL), the contents of packets denied or permitted because of the ACL are also sent to a specified NetFlow collector.

Hardware and Software Treatment of IP ACLs

ACL processing is performed in hardware. If the hardware reaches its capacity to store ACL configurations, all packets on that interface are dropped.



Note If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch or stack member, then only the traffic in that VLAN arriving on that switch is affected.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show ip access-lists hardware** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Router ACLs function as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, the flows that match a *deny* statement in a security ACL are dropped by the hardware if *ip unreachable* is disabled. The flows matching a *permit* statement are switched in hardware.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

VLAN Map Configuration Guidelines

VLAN maps are the only way to control filtering within a VLAN. VLAN maps have no direction. To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

The following are the VLAN map configuration guidelines:

- If there is no ACL configured to deny traffic on an interface and no VLAN map is configured, all traffic is permitted.
- Each VLAN map consists of a series of entries. The order of entries in a VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.
- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- Logging is not supported for VLAN maps.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, and you apply a VLAN map to a VLAN that the port belongs to, the port ACL takes precedence over the VLAN map.
- If a VLAN map configuration cannot be applied in hardware, all packets in that VLAN are dropped.

VLAN Maps with Router ACLs

To access control both bridged and routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces, and you can define a VLAN map to access control the bridged traffic.

If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.



Note When you use router ACLs with VLAN maps, packets that require logging on the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

VLAN Maps and Router ACL Configuration Guidelines

These guidelines are for configurations where you need to have a router ACL and a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- You can configure only one VLAN map and one router ACL in each direction (input/output) on a VLAN interface.
- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

```
permit... permit... permit... deny ip any any
```

or

```
deny... deny... deny... permit ip any any
```

- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.
- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

VACL Logging

When you configure VACL logging, syslog messages are generated for denied IP packets under these circumstances:

- When the first matching packet is received.
- For any matching packets received within the last 5 minutes.
- If the threshold is reached before the 5-minute interval.

Log messages are generated on a per-flow basis. A flow is defined as packets with the same IP addresses and Layer 4 (UDP or TCP) port numbers. If a flow does not receive any packets in the 5-minute interval, that flow is removed from the cache. When a syslog message is generated, the timer and packet counter are reset.

VACL logging restrictions:

- Only denied IP packets are logged.
- Packets that require logging on the outbound port ACLs are not logged if they are denied by a VACL.

Time Ranges for ACLs

You can selectively apply extended ACLs based on the time of day and the week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect,

for example, during a specified time period or on specified days of the week. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables.

These are some benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).
- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Time-based access lists trigger CPU activity because the new configuration of the access list must be merged with other features and the combined configuration loaded into the hardware memory. For this reason, you should be careful not to have several access lists configured to take affect in close succession (within a small number of minutes of each other.)



Note The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock.

IPv4 ACL Interface Considerations

When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed or are received by Layer 3 processes on the CPU. They do not affect packets bridged within a VLAN.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

By default, the input interface sends ICMP Unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

How to Configure ACLs

Configuring IPv4 ACLs

Follow the procedure given below to use IP ACLs on the switch:

SUMMARY STEPS

1. Create an ACL by specifying an access list number or name and the access conditions.
2. Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.

DETAILED STEPS

- Step 1** Create an ACL by specifying an access list number or name and the access conditions.
- Step 2** Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.

Creating a Numbered Standard ACL

Beginning in privileged EXEC mode, follow these steps to create a numbered standard ACL:

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard* [**log**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> { deny permit } <i>source source-wildcard</i> [log] Example: <pre>Switch(config)# access-list 2 deny your_host</pre>	<p>Defines a standard IPv4 access list by using a source address and wildcard.</p> <p>The <i>access-list-number</i> is a decimal number from 1 to 99 or 1300 to 1999.</p> <p>Enter deny or permit to specify whether to deny or permit access if conditions are matched.</p> <p>The <i>source</i> is the source address of the network or host from which the packet is being sent specified as:</p> <ul style="list-style-type: none"> • The 32-bit quantity in dotted-decimal format. • The keyword any as an abbreviation for <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword host as an abbreviation for source and <i>source-wildcard</i> of <i>source</i> 0.0.0.0.

	Command or Action	Purpose
		<p>(Optional) The <i>source-wildcard</i> applies wildcard bits to the source.</p> <p>(Optional) Enter log to cause an informational logging message about the packet that matches the entry to be sent to the console.</p> <p>(Optional) Enter smartlog to send copies of denied or permitted packets to a NetFlow collector.</p> <p>Note Logging is supported only on ACLs attached to Layer 3 interfaces.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Creating a Numbered Extended ACL (CLI)

Follow the procedure given below to create a numbered extended ACL:

SUMMARY STEPS

- 1. configure terminal**
- 2. access-list** *access-list-number* {deny | permit} *protocol source source-wildcard destination destination-wildcard* [precedence *precedence*] [tos *tos*] [fragments] [log [log-input]] [time-range *time-range-name*] [dscp *dscp*]
- 3. access-list** *access-list-number* {deny | permit} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [established] [precedence *precedence*] [tos *tos*] [fragments] [time-range *time-range-name*] [dscp *dscp*] [*flag*]
- 4. access-list** *access-list-number* {deny | permit} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [precedence *precedence*] [tos *tos*] [fragments] [log [log-input]] [time-range *time-range-name*] [dscp *dscp*]
- 5. access-list** *access-list-number* {deny | permit} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]]] [precedence *precedence*] [tos *tos*] [fragments] [time-range *time-range-name*] [dscp *dscp*]
- 6. access-list** *access-list-number* {deny | permit} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [precedence *precedence*] [tos *tos*] [fragments] [log [log-input]] [time-range *time-range-name*] [dscp *dscp*]
- 7. end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p>	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 2	<p>access-list <i>access-list-number</i> {deny permit} <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log</pre>	<p>Defines an extended IPv4 access list and the access conditions.</p> <p>The <i>access-list-number</i> is a decimal number from 100 to 199 or 2000 to 2699.</p> <p>Enter deny or permit to specify whether to deny or permit the packet if conditions are matched.</p> <p>For <i>protocol</i>, enter the name or number of an P protocol: ahp, eigrp, esp, gre, icmp, igmp, igrp, ip, ipinip, nos, ospf, pcp, pim, tcp, or udp, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword ip.</p> <p>Note This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see the following steps.</p> <p>The <i>source</i> is the number of the network or host from which the packet is sent.</p> <p>The <i>source-wildcard</i> applies wildcard bits to the source.</p> <p>The <i>destination</i> is the network or host number to which the packet is sent.</p> <p>The <i>destination-wildcard</i> applies wildcard bits to the destination.</p> <p>Source, source-wildcard, destination, and destination-wildcard can be specified as:</p> <ul style="list-style-type: none"> • The 32-bit quantity in dotted-decimal format. • The keyword any for 0.0.0.0 255.255.255.255 (any host). • The keyword host for a single host 0.0.0.0. <p>The other keywords are optional and have these meanings:</p> <ul style="list-style-type: none"> • precedence—Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: routine (0), priority (1), immediate (2), flash (3), flash-override (4), critical (5), internet (6), network (7). • fragments—Enter to check non-initial fragments.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • tos—Enter to match by type of service level, specified by a number from 0 to 15 or a name: normal (0), max-reliability (2), max-throughput (4), min-delay (8). • log—Enter to create an informational logging message to be sent to the console about the packet that matches the entry or log-input to include the input interface in the log entry. • time-range—Specify the time-range name. • dscp—Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values. <p>Note Your controller must support the ability to:</p> <ul style="list-style-type: none"> • Mark DCSP • Mark UP • Map DSCP and UP <p>For more information on DSCP-to-UP Mapping, see: https://tools.ietf.org/html/draft-ietf-tsvwg-ieee-802-11-01</p> <p>Note If you enter a dscp value, you cannot enter tos or precedence. You can enter both a tos and a precedence value with no dscp.</p>
Step 3	<p>access-list <i>access-list-number</i> {deny permit} tcp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [established] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>] [<i>flag</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 101 permit tcp any any eq 500</pre>	<p>Defines an extended TCP access list and the access conditions.</p> <p>The parameters are the same as those described for an extended IPv4 ACL, with these exceptions:</p> <p>(Optional) Enter an <i>operator</i> and <i>port</i> to compare source (if positioned after <i>source source-wildcard</i>) or destination (if positioned after <i>destination destination-wildcard</i>) port. Possible operators include eq (equal), gt (greater than), lt (less than), neq (not equal), and range (inclusive range). Operators require a port number (range requires two port numbers separated by a space).</p> <p>Enter the <i>port</i> number as a decimal number (from 0 to 65535) or the name of a TCP port. Use only TCP port numbers or names when filtering TCP.</p> <p>The other optional keywords have these meanings:</p>

	Command or Action	Purpose
		<ul style="list-style-type: none"> • established—Enter to match an established connection. This has the same function as matching on the ack or rst flag. • flag—Enter one of these flags to match by the specified TCP header bits: ack (acknowledge), fin (finish), psh (push), rst (reset), syn (synchronize), or urg (urgent).
Step 4	<p>access-list <i>access-list-number</i> {deny permit} udp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [precedence precedence] [tos tos] [fragments] [log [log-input]] [time-range <i>time-range-name</i>] [dscp dscp]</p> <p>Example:</p> <pre>Switch(config)# access-list 101 permit udp any any eq 100</pre>	<p>(Optional) Defines an extended UDP access list and the access conditions.</p> <p>The UDP parameters are the same as those described for TCP except that the [<i>operator</i> [<i>port</i>]] port number or name must be a UDP port number or name, and the flag and established keywords are not valid for UDP.</p>
Step 5	<p>access-list <i>access-list-number</i> {deny permit} icmp <i>source source-wildcard destination destination-wildcard</i> [<i>icmp-type</i>] [[<i>icmp-type icmp-code</i>] [<i>icmp-message</i>]] [precedence precedence] [tos tos] [fragments] [time-range <i>time-range-name</i>] [dscp dscp]</p> <p>Example:</p> <pre>Switch(config)# access-list 101 permit icmp any any 200</pre>	<p>Defines an extended ICMP access list and the access conditions.</p> <p>The ICMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • <i>icmp-type</i>—Enter to filter by ICMP message type, a number from 0 to 255. • <i>icmp-code</i>—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • <i>icmp-message</i>—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name.
Step 6	<p>access-list <i>access-list-number</i> {deny permit} igmp <i>source source-wildcard destination destination-wildcard</i> [<i>igmp-type</i>] [precedence precedence] [tos tos] [fragments] [log [log-input]] [time-range <i>time-range-name</i>] [dscp dscp]</p> <p>Example:</p> <pre>Switch(config)# access-list 101 permit igmp any any 14</pre>	<p>(Optional) Defines an extended IGMP access list and the access conditions.</p> <p>The IGMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with this optional parameter.</p> <p><i>igmp-type</i>—To match IGMP message type, enter a number from 0 to 15, or enter the message name: dvmrp, host-query, host-report, pim, or trace.</p>
Step 7	<p>end</p> <p>Example:</p>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
	Switch(config)# end	

Creating Named Standard ACLs

Follow the procedure given below to create a standard ACL using names:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list standard *name***
4. Use one of the following:
 - **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
 - **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip access-list standard <i>name</i> Example: Switch(config)# ip access-list standard 20	Defines a standard IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 1 to 99.
Step 4	Use one of the following: <ul style="list-style-type: none"> • deny {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] • permit {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] 	In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped. <ul style="list-style-type: none"> • host <i>source</i>—A source and source wildcard of <i>source</i> 0.0.0.0.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255</pre> <p>or</p> <pre>Switch(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0</pre>	<ul style="list-style-type: none"> • any—A source and source wildcard of 0.0.0.0 255.255.255.255.
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-std-nacl)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating Extended Named ACLs

Follow the procedure given below to create an extended ACL using names:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. **{deny | permit} protocol {source [source-wildcard] | host source | any} {destination [destination-wildcard] | host destination | any} [precedence precedence] [tos tos] [established] [log] [time-range time-range-name]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip access-list extended <i>name</i> Example: Switch(config)# ip access-list extended 150	Defines an extended IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 100 to 199.
Step 4	{deny permit} protocol {source [source-wildcard] host source any} {destination [destination-wildcard] host destination any} [precedence precedence] [tos tos] [established] [log] [time-range time-range-name] Example: Switch(config-ext-nacl)# permit 0 any any	In access-list configuration mode, specify the conditions allowed or denied. Use the log keyword to get access list logging messages, including violations. <ul style="list-style-type: none"> • host source—A source and source wildcard of <i>source</i> 0.0.0.0. • host destination—A destination and destination wildcard of <i>destination</i> 0.0.0.0. • any—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 5	end Example: Switch(config-ext-nacl)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

When you are creating extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL.

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to do next

After creating a named ACL, you can apply it to interfaces or to VLANs.

Configuring Time Ranges for ACLs

Follow these steps to configure a time-range parameter for an ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **time-range** *time-range-name*
4. Use one of the following:
 - **absolute** [**start** *time date*] [**end** *time date*]
 - **periodic** *day-of-the-week hh:mm to [day-of-the-week] hh:mm*
 - **periodic** {*weekdays* | *weekend* | **daily**} *hh:mm to hh:mm*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch(config)# enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	time-range <i>time-range-name</i> Example:	Assigns a meaningful name (for example, <i>workhours</i>) to the time range to be created, and enter time-range

	Command or Action	Purpose
	Switch(config) # time-range workhours	configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
Step 4	<p>Use one of the following:</p> <ul style="list-style-type: none"> • absolute [start time date] [end time date] • periodic day-of-the-week hh:mm to [day-of-the-week] hh:mm • periodic {weekdays weekend daily} hh:mm to hh:mm <p>Example:</p> <pre>Switch(config-time-range) # absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006</pre> <p>or</p> <pre>Switch(config-time-range) # periodic weekdays 8:00 to 12:00</pre>	<p>Specifies when the function it will be applied to is operational.</p> <ul style="list-style-type: none"> • You can use only one absolute statement in the time range. If you configure more than one absolute statement, only the one configured last is executed. • You can enter multiple periodic statements. For example, you could configure different hours for weekdays and weekends. <p>See the example configurations.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config) # end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Repeat the steps if you have multiple items that you want in effect at different times.

Applying an IPv4 ACL to a Terminal Line

You can use numbered ACLs to control access to one or more terminal lines. You cannot apply named ACLs to lines. You must set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them.

Follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line [console | vty] line-number**
4. **access-class access-list-number {in | out}**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch(config)# enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	line [console vty] line-number Example: <pre>Switch(config)# line console 0</pre>	Identifies a specific line to configure, and enter in-line configuration mode. <ul style="list-style-type: none"> • console—Specifies the console terminal line. The console port is DCE. • vtty—Specifies a virtual terminal for remote console access. <p>The <i>line-number</i> is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.</p>
Step 4	access-class access-list-number {in out} Example: <pre>Switch(config-line)# access-class 10 in</pre>	Restricts incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config-line)# end	
Step 6	show running-config Example: Switch# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Applying an IPv4 ACL to an Interface (CLI)

This section describes how to apply IPv4 ACLs to network interfaces.

Beginning in privileged EXEC mode, follow the procedure given below to control access to an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip access-group** {*access-list-number* | *name*} {**in** | **out**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Identifies a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL), or a Layer 3 interface (router ACL).
Step 3	ip access-group { <i>access-list-number</i> <i>name</i> } { in out }	Controls access to the specified interface.

	Command or Action	Purpose
	Device(config-if)# <code>ip access-group 2 in</code>	
Step 4	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Displays the access list configuration.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Creating Named MAC Extended ACLs

You can filter non-IPv4 traffic on a VLAN or on a Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs.

Follow these steps to create a named MAC extended ACL:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `mac access-list extended name`
4. `{deny | permit} {any | host source MAC address | source MAC address mask} {any | host destination MAC address | destination MAC address mask} [type mask | lsap lsap mask | aarp | amber | dec-spanning | decnet-iv | diagnostic | dsm | etype-6000 | etype-8042 | lat | larc-sca | mop-console | mop-dump | msdos | mumps | netbios | vines-echo | vines-ip | xns-idp | 0-65535] [cos cos]`
5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	mac access-list extended name Example: Switch(config)# mac access-list extended macl	Defines an extended MAC access list using a name.
Step 4	{deny permit} {any host source MAC address source MAC address mask} {any host destination MAC address destination MAC address mask} [type mask lsap lsap mask aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat larc-sca mop-console mop-dump msdos mumps netbios vines-echo vines-ip xns-idp 0-65535] [cos cos] Example: Switch(config-ext-macl)# deny any any decnet-iv or Switch(config-ext-macl)# permit any any	In extended MAC access-list configuration mode, specifies to permit or deny any source MAC address, a source MAC address with a mask, or a specific host source MAC address and any destination MAC address, destination MAC address with a mask, or a specific destination MAC address. (Optional) You can also enter these options: <ul style="list-style-type: none"> • <i>type mask</i>—An arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hexadecimal, or octal with optional mask of <i>don't care</i> bits applied to the EtherType before testing for a match. • lsap lsap mask—An LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hexadecimal, or octal with optional mask of <i>don't care</i> bits. • aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat larc-sca mop-console mop-dump msdos mumps netbios vines-echo vines-ip xns-idp—A non-IP protocol. • cos cos—An IEEE 802.1Q cost of service number from 0 to 7 used to set priority.
Step 5	end Example: Switch(config-ext-macl)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show running-config</code>	
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Applying a MAC ACL to a Layer 2 Interface

Follow these steps to apply a MAC access list to control access to a Layer 2 interface:

SUMMARY STEPS

1. **configure terminal**
2. **configure terminal**
3. **interface *interface-id***
4. **mac access-group {*name*} {in }**
5. **end**
6. **show mac access-group [interface *interface-id*]**
7. **configure terminal**
8. **configure terminal**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet1/0/2</code>	Identifies a specific interface, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).

	Command or Action	Purpose
Step 4	mac access-group { <i>name</i> } { in } Example: Switch(config-if)# mac access-group mac1 in	Controls access to the specified interface by using the MAC access list. Port ACLs are supported in the outbound and inbound directions .
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 6	show mac access-group [interface <i>interface-id</i>] Example: Switch# show mac access-group interface gigabitethernet1/0/2	Displays the MAC access list applied to the interface or all Layer 2 interfaces.
Step 7	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 8	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

After receiving a packet, the switch checks it against the inbound ACL. If the ACL permits it, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it. When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Configuring VLAN Maps

Follow the procedure given below to create a VLAN map and apply it to one or more VLANs:

Before you begin

Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN.

SUMMARY STEPS

1. **vlan access-map** *name* [**number**]
2. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]

- Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended):

- **action { forward}**

```
Switch(config-access-map)# action forward
```

- **action { drop}**

```
Switch(config-access-map)# action drop
```

- vlan filter** *mapname* **vlan-list** *list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>vlan access-map <i>name</i> [<i>number</i>]</p> <p>Example:</p> <pre>Switch(config)# vlan access-map map_1 20</pre>	<p>Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.</p> <p>Entering this command changes to access-map configuration mode.</p>
Step 2	<p>match {ip mac} address {<i>name</i> <i>number</i>} [<i>name</i> <i>number</i>]</p> <p>Example:</p> <pre>Switch(config-access-map)# match ip address ip2</pre>	<p>Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.</p> <p>Note If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, all IP and Layer 2 packets are dropped.</p>
Step 3	Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address)	Sets the action for the map entry.

	Command or Action	Purpose
	<p>and to match the packet against one or more ACLs (standard or extended):</p> <ul style="list-style-type: none"> • action { forward} <pre>Switch(config-access-map) # action forward</pre> <ul style="list-style-type: none"> • action { drop} <pre>Switch(config-access-map) # action drop</pre>	
Step 4	<p>vlan filter <i>mapname</i> vlan-list <i>list</i></p> <p>Example:</p> <pre>Switch(config) # vlan filter map 1 vlan-list 20-22</pre>	<p>Applies the VLAN map to one or more VLAN IDs.</p> <p>The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.</p>

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries. Beginning in privileged EXEC mode, follow these steps to create, add to, or delete a VLAN map entry:

SUMMARY STEPS

1. **configure terminal**
2. **vlan access-map** *name* [**number**]
3. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
4. **action** {**drop** | **forward**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>

	Command or Action	Purpose
Step 2	<p>vlan access-map <i>name</i> [<i>number</i>]</p> <p>Example:</p> <pre>Switch(config)# vlan access-map map_1 20</pre>	<p>Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.</p> <p>Entering this command changes to access-map configuration mode.</p>
Step 3	<p>match {<i>ip</i> <i>mac</i>} <i>address</i> {<i>name</i> <i>number</i>} [<i>name</i> <i>number</i>]</p> <p>Example:</p> <pre>Switch(config-access-map)# match ip address ip2</pre>	<p>Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.</p>
Step 4	<p>action {<i>drop</i> <i>forward</i>}</p> <p>Example:</p> <pre>Switch(config-access-map)# action forward</pre>	<p>(Optional) Sets the action for the map entry. The default is to forward.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config-access-map)# end</pre>	<p>Returns to global configuration mode.</p>
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Displays the access list configuration.</p>
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Applying a VLAN Map to a VLAN

To apply a VLAN map to one or more VLANs, perform these steps.

SUMMARY STEPS

- 1.
2. **configure terminal**
3. **vlan filter *mapname* vlan-list *list***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1		
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vlan filter <i>mapname</i> vlan-list <i>list</i> Example: Switch(config)# vlan filter map 1 vlan-list 20-22	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Displays the access list configuration.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring VACL Logging

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **vlan access-map name** [*number*]
3. **action drop log**
4. **exit**
5. **vlan access-log** {**maxflow** *max_number* | **threshold** *pkt_count*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	vlan access-map name [<i>number</i>] Example: <pre>Switch(config)# vlan access-map gandymede 10</pre>	<p>Creates a VLAN map. Give it a name and optionally a number. The number is the sequence number of the entry within the map.</p> <p>The sequence number range is from 0 to 65535.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>Specifying the map name and optionally a number enters the access-map configuration mode.</p>
Step 3	action drop log Example: <pre>Switch(config-access-map)# action drop log</pre>	Sets the VLAN access map to drop and log IP packets.
Step 4	exit Example: <pre>Switch(config-access-map)# exit</pre>	Exits the VLAN access map configuration mode and return to the global configuration mode.
Step 5	vlan access-log { maxflow <i>max_number</i> threshold <i>pkt_count</i> } Example:	<p>Configures the VACL logging parameters.</p> <ul style="list-style-type: none"> • maxflow <i>max_number</i>—Sets the log table size. The content of the log table can be deleted by setting the

	Command or Action	Purpose
	Switch(config)# vlan access-log threshold 4000	<p>maxflow to 0. When the log table is full, the software drops logged packets from new flows.</p> <p>The range is from 0 to 2048. The default is 500.</p> <ul style="list-style-type: none"> • threshold <i>pkt_count</i>—Sets the logging threshold. A logging message is generated if the threshold for a flow is reached before the 5-minute interval. <p>The threshold range is from 0 to 2147483647. The default threshold is 0, which means that a syslog message is generated every 5 minutes.</p>
Step 6	<p>end</p> <p>Example:</p> <p>Switch(config)# end</p>	Returns to privileged EXEC mode.

Monitoring IPv4 ACLs

You can monitor IPv4 ACLs by displaying the ACLs that are configured on the switch, and displaying the ACLs that have been applied to interfaces and VLANs.

When you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in this table to display this information.

Table 133: Commands for Displaying Access Lists and Access Groups

Command	Purpose
show access-lists [<i>number</i> <i>name</i>]	Displays the contents of one or all current IP and MAC address access lists (numbered or named).
show ip access-lists [<i>number</i> <i>name</i>]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).
show ip interface <i>interface-id</i>	Displays detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the ip access-group configuration command, the access groups are included in the display.
show running-config [interface <i>interface-id</i>]	Displays the contents of the configuration file for the switch or the interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [interface <i>interface-id</i>]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.

Configuration Examples for ACLs

Examples: Using Time Ranges with ACLs

This example shows how to verify after you configure time ranges for *workhours* and to configure January 1, 2006, as a company holiday.

```
Switch# show time-range
time-range entry: new_year_day_2003 (inactive)
  absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
  periodic weekdays 8:00 to 12:00
  periodic weekdays 13:00 to 17:00
```

To apply a time range, enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday times and permits all TCP traffic during work hours.

```
Switch(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Switch(config)# access-list 188 permit tcp any any time-range workhours
Switch(config)# end
Switch# show access-lists
Extended IP access list 188
  10 deny tcp any any time-range new_year_day_2006 (inactive)
  20 permit tcp any any time-range workhours (inactive)
```

This example uses named ACLs to permit and deny the same traffic.

```
Switch(config)# ip access-list extended deny_access
Switch(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Switch(config-ext-nacl)# exit
Switch(config)# ip access-list extended may_access
Switch(config-ext-nacl)# permit tcp any any time-range workhours
Switch(config-ext-nacl)# end
Switch# show ip access-lists
Extended IP access list lpip_default
  10 permit ip any any
Extended IP access list deny_access
  10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
  10 permit tcp any any time-range workhours (inactive)
```

Examples: Including Comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would

be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

In this example, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith through
Switch(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

Examples: Troubleshooting ACLs

If this ACL manager message appears and [chars] is the access-list name,

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

The switch has insufficient resources to create a hardware representation of the ACL. The resources include hardware memory and label space but not CPU memory. A lack of available logical operation units or specialized hardware resources causes this problem. Logical operation units are needed for a TCP flag match or a test other than **eq** (**ne**, **gt**, **lt**, or **range**) on TCP, UDP, or SCTP port numbers.

Use one of these workarounds:

- Modify the ACL configuration to use fewer resources.
- Rename the ACL with a name or number that alphanumerically precedes the ACL names or numbers.

To determine the specialized hardware resources, enter the **show platform layer4 acl** map privileged EXEC command. If the switch does not have available resources, the output shows that index 0 to index 15 are not available.

For more information about configuring ACLs with insufficient resources, see CSCsq63926 in the Bug Toolkit.

For example, if you apply this ACL to an interface:

```
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
permit tcp source source-wildcard destination destination-wildcard
```

And if this message appears:

ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]

The flag-related operators are not available. To avoid this issue,

- Move the fourth ACE before the first ACE by using **ip access-list resequence** global configuration command:

```
permit tcp source source-wildcard destination destination-wildcard
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
```

or

- Rename the ACL with a name or number that alphanumerically precedes the other ACLs (for example, rename ACL 79 to ACL 1).

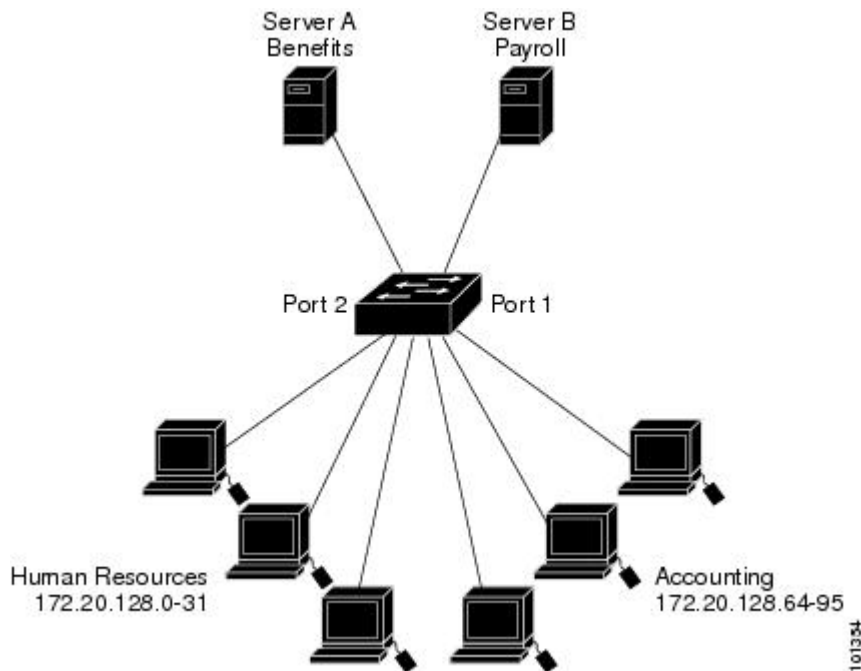
You can now apply the first ACE in the ACL to the interface. The switch allocates the ACE to available mapping bits in the Opselect index and then allocates flag-related operators to use the same bits in the hardware memory.

IPv4 ACL Configuration Examples

This section provides examples of configuring and applying IPv4 ACLs. For detailed information about compiling ACLs, see the *Cisco IOS Security Configuration Guide, Release 12.4* and to the “Configuring IP Services” section in the “IP Addressing and Services” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

ACLs in a Small Networked Office

Figure 96: Using Router ACLs to Control Traffic



This shows a small networked office environment with routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing confidential payroll data. All users can access Server A, but Server B has restricted access.

Use router ACLs to do this in one of two ways:

- Create a standard ACL, and filter traffic coming to the server from Port 1.
- Create an extended ACL, and filter traffic coming from the server into Port 1.

Examples: ACLs in a Small Networked Office

This example uses a standard ACL to filter traffic coming into Server B from a port, permitting traffic only from Accounting's source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic coming out of routed Port 1 from the specified source address.

```
Switch(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Standard IP access list 6
 10 permit 172.20.128.64, wildcard bits 0.0.0.31
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic coming from Server B into a port, permitting traffic from any source address (in this case Server B) to only the Accounting destination addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic going into routed Port 1, permitting it to go only to the specified

destination addresses. Note that with extended ACLs, you must enter the protocol (IP) before the source and destination information.

```
Switch(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Extended IP access list 106
    10 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 106 in
```

Example: Numbered ACLs

In this example, network 10.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 10.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 10.0.0.0 subnets. The ACL is applied to packets entering a port.

```
Switch(config)# access-list 2 permit 10.48.0.3
Switch(config)# access-list 2 deny 10.48.0.0 0.0.255.255
Switch(config)# access-list 2 permit 10.0.0.0 0.255.255.255
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ip access-group 2 in
```

Examples: Extended ACLs

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# access-list 102 permit icmp any any
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ip access-group 102 in
```

In this example, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system of the network always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Switch(config)# interface gigabitethernet1/0/1
```

```
Switch(config-if)# ip access-group 102 in
```

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 1 on stack member 1 is the interface that connects the router to the Internet.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 102 in
```

Examples: Named ACLs

Creating named standard and extended ACLs

This example creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Switch(config)# ip access-list standard Internet_filter
Switch(config-ext-nacl)# permit 1.2.3.4
Switch(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Switch(config)# ip access-list extended marketing_group
Switch(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Switch(config-ext-nacl)# deny tcp any any
Switch(config-ext-nacl)# permit icmp any any
Switch(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Switch(config-ext-nacl)# deny ip any any log
Switch(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Switch(config)# interface gigabitethernet3/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 2.0.5.1 255.255.255.0
Switch(config-if)# ip access-group Internet_filter out
Switch(config-if)# ip access-group marketing_group in
```

Deleting individual ACEs from named ACLs

This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Switch(config)# ip access-list extended border-list
```



```
Switch(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Examples: Time Range Applied to an IP ACL

This example denies HTTP traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m. (18:00). The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m. (20:00).

```
Switch(config)# time-range no-http
Switch(config)# periodic weekdays 8:00 to 18:00
!
Switch(config)# time-range udp-yes
Switch(config)# periodic weekend 12:00 to 20:00
!
Switch(config)# ip access-list extended strict
Switch(config-ext-nacl)# deny tcp any any eq www time-range no-http
Switch(config-ext-nacl)# permit udp any any time-range udp-yes
!
Switch(config-ext-nacl)# exit
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ip access-group strict in
```

Examples: Configuring Commented IP ACL Entries

In this example of a numbered ACL, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith workstation through
Switch(config)# access-list 1 deny 171.69.3.13
```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the web:

```
Switch(config)# access-list 100 remark Do not allow Winter to browse the web
Switch(config)# access-list 100 deny host 171.69.3.85 any eq www
Switch(config)# access-list 100 remark Do not allow Smith to browse the web
Switch(config)# access-list 100 deny host 171.69.3.13 any eq www
```

In this example of a named ACL, the Jones subnet is not allowed access:

```
Switch(config)# ip access-list standard prevention
Switch(config-std-nacl)# remark Do not allow Jones subnet through
Switch(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

Examples: ACL Logging

Two variations of logging are supported on ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.

```
Switch(config)# ip access-list standard stan1
Switch(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Switch(config-std-nacl)# permit any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group stan1 in
Switch(config-if)# end
Switch# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
  Console logging: level debugging, 37 messages logged
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 37 messages logged
  File logging: disabled
  Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets.

```
Switch(config)# ip access-list extended ext1
Switch(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Switch(config-ext-nacl)# deny udp any any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip access-group ext1 in
```

This is an example of a log for an extended ACL:

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1 packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8 packets
```

Note that all logging entries for IP ACLs start with %SEC-6-IPACCESSLOG with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
->
10.1.1.61 (0/0), 1 packet
```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
packet
```

Configuration Examples for ACLs and VLAN Maps

Example: Creating an ACL and a VLAN Map to Deny a Packet

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the *ip1* ACL (TCP packets) would be dropped. You first create the *ip1* ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Switch(config)# ip access-list extended ip1
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 10
Switch(config-access-map)# match ip address ip1
Switch(config-access-map)# action drop
```

Example: Creating an ACL and a VLAN Map to Permit a Packet

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the *ip2* ACL are forwarded. In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

```
Switch(config)# ip access-list extended ip2
Switch(config-ext-nacl)# permit udp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 20
Switch(config-access-map)# match ip address ip2
Switch(config-access-map)# action forward
```

Example: Default Action of Dropping IP Packets and Forwarding MAC Packets

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists **igmp-match** and **tcp-match**, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets
- Drop all other IP packets

- Forward all non-IP packets

```
Switch(config)# access-list 101 permit udp any any
Switch(config)# ip access-list extended igmp-match
Switch(config-ext-nacl)# permit igmp any any

Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-ip-default 10
Switch(config-access-map)# match ip address 101
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 20
Switch(config-access-map)# match ip address igmp-match
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 30
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
```

Example: Default Action of Dropping MAC Packets and Forwarding IP Packets

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists **good-hosts** and **good-protocols**, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets
- Forward all IP packets

Example: Default Action of Dropping All Packets

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists **tcp-match** and **good-hosts** from Examples 2 and 3, the map will have the following results:

- Forward all TCP packets
- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Drop all other IP packets
- Drop all other MAC packets

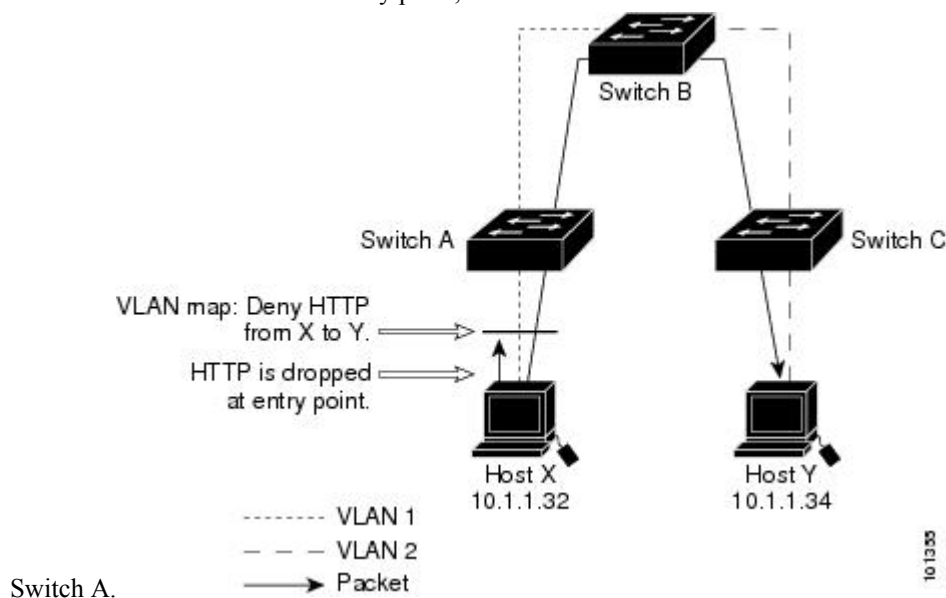
```
Switch(config)# vlan access-map drop-all-default 10
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-all-default 20
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
```

Configuration Examples for Using VLAN Maps in Your Network

Example: Wiring Closet Configuration

Figure 97: Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. Assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, a Layer 3 switch with routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point,



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not bridge it to Switch B.

First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```
Switch(config)# ip access-list extended http
Switch(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Switch(config-ext-nacl)# exit
```

Next, create VLAN access map *map2* so that traffic that matches the *http* access list is dropped and all other IP traffic is forwarded.

```
Switch(config)# vlan access-map map2 10
Switch(config-access-map)# match ip address http
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# ip access-list extended match_all
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map2 20
Switch(config-access-map)# match ip address match_all
```

```
Switch(config-access-map)# action forward
```

Then, apply VLAN access map *map2* to VLAN 1.

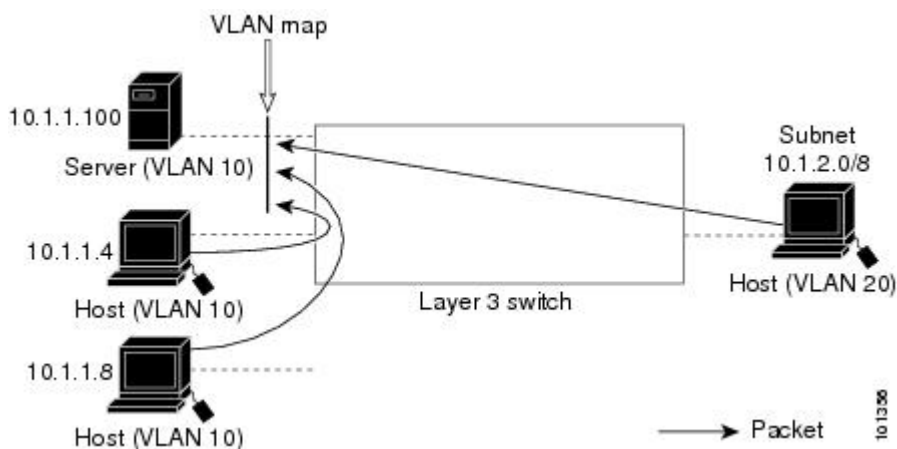
```
Switch(config)# vlan filter map2 vlan 1
```

Example: Restricting Access to a Server on Another VLAN

Figure 98: Restricting Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access denied to these hosts:

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.



Example: Denying Access to a Server on Another VLAN

This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER1_MAP that denies access to hosts in subnet 10.1.2.0.8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1_MAP to VLAN 10.

Define the IP ACL that will match the correct packets.

```
Switch(config)# ip access-list extended SERVER1_ACL
Switch(config-ext-nacl)# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Switch(config-ext-nacl)# permit ip host 10.1.1.4 host 10.1.1.100
Switch(config-ext-nacl)# permit ip host 10.1.1.8 host 10.1.1.100
Switch(config-ext-nacl)# exit
```

Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```
Switch(config)# vlan access-map SERVER1_MAP
Switch(config-access-map)# match ip address SERVER1_ACL
Switch(config-access-map)# action drop
Switch(config)# vlan access-map SERVER1_MAP 20
```

```
Switch(config-access-map) # action forward
Switch(config-access-map) # exit
```

Apply the VLAN map to VLAN 10.

```
Switch(config) # vlan filter SERVER1_MAP vlan-list 10
```

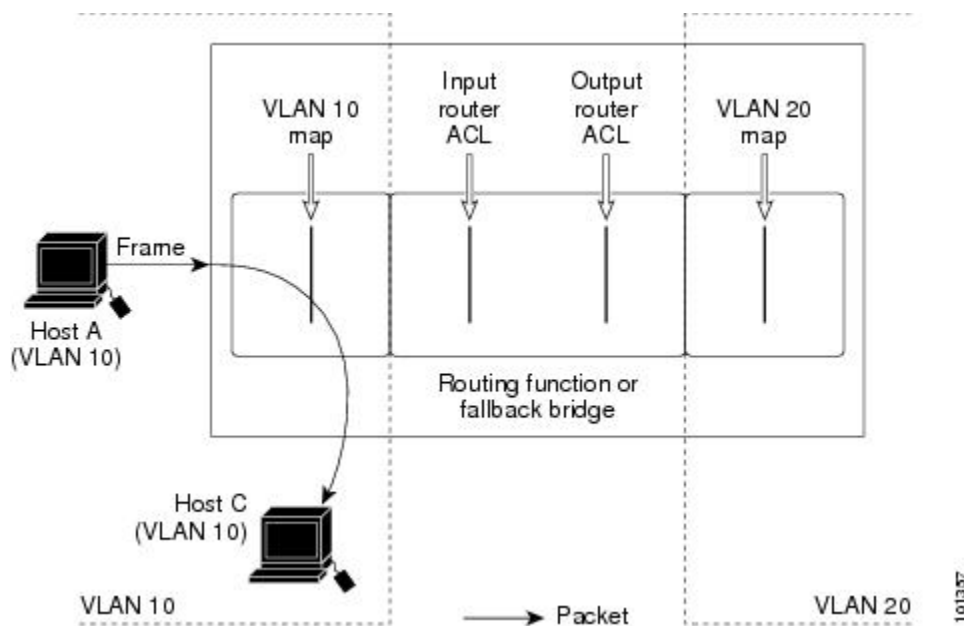
Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, bridged, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

Example: ACLs and Switched Packets

Figure 99: Applying ACLs on Switched Packets

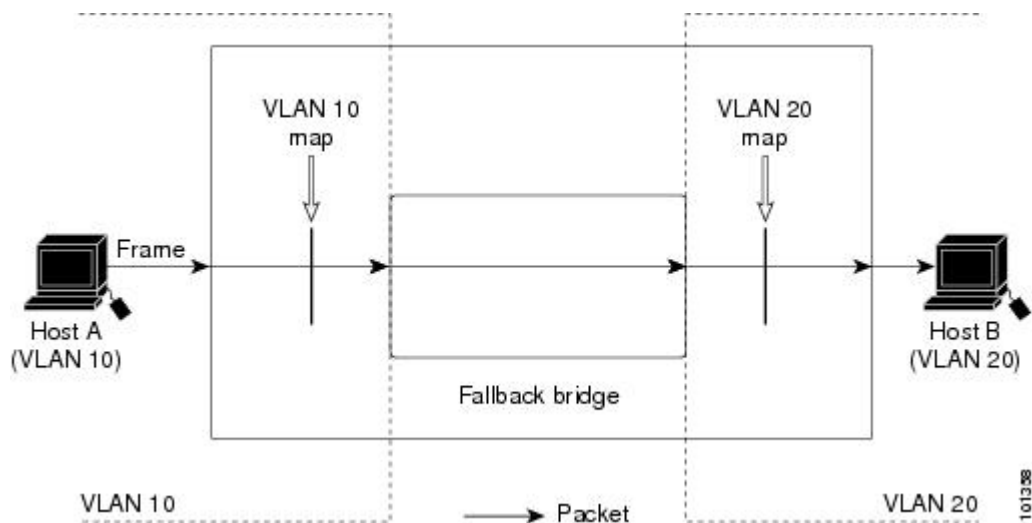
This example shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded by fallback bridging are only subject to the VLAN map of the input VLAN.



Example: ACLs and Bridged Packets

Figure 100: Applying ACLs on Bridged Packets

This example shows how an ACL is applied on fallback-bridged packets. For bridged packets, only Layer 2 ACLs are applied to the input VLAN. Only non-IP, non-ARP packets can be fallback-bridged.

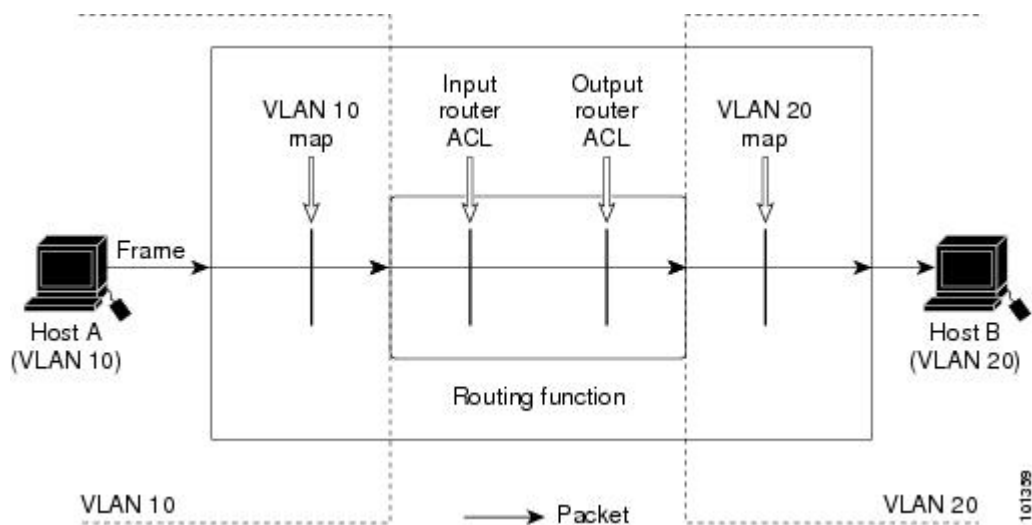


Example: ACLs and Routed Packets

Figure 101: Applying ACLs on Routed Packets

This example shows how ACLs are applied on routed packets. The ACLs are applied in this order:

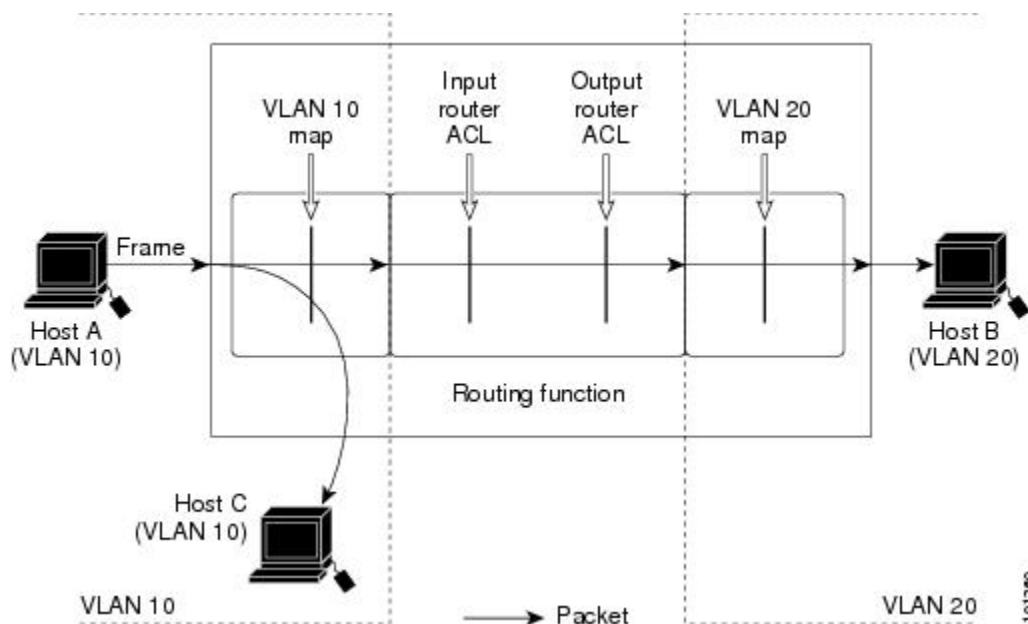
1. VLAN map for input VLAN
2. Input router ACL
3. Output router ACL
4. VLAN map for output VLAN



Example: ACLs and Multicast Packets

Figure 102: Applying ACLs on Multicast Packets

This example shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN. The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map drops the packet, no destination receives a copy of the packet.



Feature Information for IPv4 Access Control Lists

Release	Feature Information
Cisco IOS Release 15.2(3)E	IPv4 Access Control Lists perform packet filtering to control which packets move through the network and where. Such control provides security by helping to limit network traffic, restrict the access of users and devices to the network, and prevent traffic from leaving a network. This feature was introduced.
Cisco IOS 15.2(2)E	The Named ACL Support for Noncontiguous Ports on an Access Control Entry feature allows you to specify noncontiguous ports in a single access control entry, which greatly reduces the number of entries required in an access control list when several entries have the same source address, destination address, and protocol, but differ only in the ports.

Release	Feature Information
Cisco IOS 15.2(2)E	<p>The IP Access List Entry Sequence Numbering feature helps users to apply sequence numbers to permit or deny statements and also reorder, add, or remove such statements from a named IP access list. This feature makes revising IP access lists much easier. Prior to this feature, users could add access list entries to the end of an access list only; therefore needing to add statements anywhere except the end required reconfiguring the access list entirely.</p> <p>The following commands were introduced or modified: deny (IP), ip access-list resequence deny (IP), permit (IP).</p>



CHAPTER 64

IPv6 ACLs

- [Finding Feature Information, on page 1267](#)
- [IPv6 ACLs Overview, on page 1267](#)
- [Restrictions for IPv6 ACLs, on page 1268](#)
- [Default Configuration for IPv6 ACLs , on page 1269](#)
- [Configuring IPv6 ACLs, on page 1269](#)
- [Attaching an IPv6 ACL to an Interface, on page 1273](#)
- [Monitoring IPv6 ACLs, on page 1274](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

IPv6 ACLs Overview

You can filter IP Version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similar to how you create and apply IP Version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic when the switch is running IP base and LAN base feature sets.

A switch supports three types of IPv6 ACLs:

- IPv6 router ACLs are supported on outbound or inbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. IPv6 router ACLs apply only to IPv6 packets that are routed.
- IPv6 port ACLs are supported on outbound and inbound Layer 2 interfaces. IPv6 port ACLs are applied to all IPv6 packets entering the interface.
- VLAN ACLs or VLAN maps access-control all packets in a VLAN. You can use VLAN maps to filter traffic between devices in the same VLAN. ACL VLAN maps are applied on L2 VLANs. VLAN maps

are configured to provide access control based on Layer 3 addresses for IPv6. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets entering the VLAN are checked against the VLAN map.

The switch supports VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface. As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs.

Interactions with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface. If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.

- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, packets are dropped on the interface and an unload error message is logged.

Restrictions for IPv6 ACLs

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most Cisco IOS-supported IPv6 ACLs with some exceptions:

- The switch does not support matching on these keywords: **routing header**, and **undetermined-transport**.
- The switch does not support reflexive ACLs (the **reflect** keyword).
- This release supports port ACLs, router ACLs and VLAN ACLs (VLAN maps) for IPv6.
- Output router ACLs and input port ACLs for IPv6 are supported only on switch stacks. Switches support only control plane (incoming) IPv6 ACLs.
- The switch does not apply MAC-based ACLs on IPv6 frames.
- When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.

- If an ACL is applied to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the ACL that is currently attached to the interface.

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the **fragments** keyword as in IPv4) are supported
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of hardware space, the packets associated with the ACL are dropped on the interface.
- Routed or bridged packets with hop-by-hop options have IPv6 ACLs applied in software.
- Logging is supported for router ACLs, but not for port ACLs.
- The switch supports IPv6 address-matching for a full range of prefix-lengths.
- If a downloadable ACL contains any type of duplicate entries, the entries are not auto merged. As a result, the 802.1X session authorization fails. Ensure that the downloadable ACL is optimized without any duplicate entries, for example port-based and name-based entries for the same port.

Default Configuration for IPv6 ACLs

The default IPv6 ACL configuration is as follows:

```
Switch# show access-lists preauth_ipv6_acl
IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

Configuring IPv6 ACLs

To filter IPv6 traffic, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **{ipv6 access-list list-name**
4. **{deny | permit} protocol {source-ipv6-prefix/|prefix-length |any| host source-ipv6-address} [operator [port-number]] { destination-ipv6-prefix/ prefix-length | any | host destination-ipv6-address} [operator [port-number]][dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]**

5. **{deny | permit} tcp** {source-ipv6-prefix/prefix-length | **any** | **host** source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | **any** | **host** destination-ipv6-address} [operator [port-number]] [**ack**] [**dscp** value] [**established**] [**fin**] [**log**] [**log-input**] [**neq** {port | protocol}] [**psh**] [**range** {port | protocol}] [**rst**] [**routing**] [**sequence** value] [**syn**] [**time-range** name] [**urg**]
6. **{deny | permit} udp** {source-ipv6-prefix/prefix-length | **any** | **host** source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | **any** | **host** destination-ipv6-address} [operator [port-number]] [**dscp** value] [**log**] [**log-input**] [**neq** {port | protocol}] [**range** {port | protocol}] [**routing**] [**sequence** value] [**time-range** name]
7. **{deny | permit} icmp** {source-ipv6-prefix/prefix-length | **any** | **host** source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | **any** | **host** destination-ipv6-address} [operator [port-number]] [**icmp-type** [icmp-code] | icmp-message] [**dscp** value] [**log**] [**log-input**] [**routing**] [**sequence** value] [**time-range** name]
8. **end**
9. **show ipv6 access-list**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	{ipv6 access-list list-name} Example: Switch(config)# ipv6 access-list example_acl_list	Defines an IPv6 ACL name, and enters IPv6 access list configuration mode.
Step 4	{deny permit} protocol {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: <ul style="list-style-type: none"> • For protocol, enter the name or number of an IP: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number. • The <i>source-ipv6-prefix/prefix-length</i> or <i>destination-ipv6-prefix/prefix-length</i> is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in

	Command or Action	Purpose
		<p>hexadecimal and using 16-bit values between colons (see RFC 2373).</p> <ul style="list-style-type: none"> • Enter any as an abbreviation for the IPv6 prefix <code>::/0</code>. • For host <i>source-ipv6-address</i> or <i>destination-ipv6-address</i>, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range. If the operator follows the <i>source-ipv6-prefix/prefix-length</i> argument, it must match the source port. If the operator follows the <i>destination-ipv6-prefix/prefix-length</i> argument, it must match the destination port. • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6. • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4,294,967,295. • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.

	Command or Action	Purpose
Step 5	<pre>{deny permit} tcp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6- prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [ack] [dscp value] [established] [fin] [log] [log-input] [neq {port protocol}] [psh] [range {port protocol}] [rst] [routing] [sequence value] [syn] [time-range name] [urg]</pre>	<p>(Optional) Define a TCP access list and the access conditions.</p> <p>Enter tcp for Transmission Control Protocol. The parameters are the same as those described in Step 3a, with these additional optional parameters:</p> <ul style="list-style-type: none"> • ack: Acknowledgment bit set. • established: An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. • fin: Finished bit set; no more data from sender. • neq {port protocol}: Matches only packets that are not on a given port number. • psh—Push function bit set. • range {port protocol}: Matches only packets in the port number range. • rst: Reset bit set. • syn: Synchronize bit set. • urg: Urgent pointer bit set.
Step 6	<pre>{deny permit} udp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [dscp value] [log] [log-input] [neq {port protocol}] [range {port protocol}] [routing] [sequence value] [time-range name]]</pre>	<p>(Optional) Define a UDP access list and the access conditions.</p> <p>Enter udp for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the [operator [port]] port number or name must be a UDP port number or name, and the established parameter is not valid for UDP.</p>
Step 7	<pre>{deny permit} icmp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [icmp-type [icmp-code] icmp-message] [dscp value] [log] [log-input] [routing] [sequence value] [time-range name]</pre>	<p>(Optional) Define an ICMP access list and the access conditions.</p> <p>Enter icmp for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 1, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • icmp-type: Enter to filter by ICMP message type, a number from 0 to 255. • icmp-code: Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • icmp-message: Enter to filter ICMP packets by the ICMP message type name or the ICMP message type

	Command or Action	Purpose
		and code name. To see a list of ICMP message type names and code names, use the ? key or see command reference for this release.
Step 8	end	Return to privileged EXEC mode.
Step 9	show ipv6 access-list	Verify the access list configuration.
Step 10	show running-config Example: Switch# show running-config	Verifies your entries.
Step 11	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Attaching an IPv6 ACL to an Interface

You can apply an ACL to outbound or inbound traffic on Layer 3 interfaces, or to inbound traffic on Layer 2 interfaces. You can also apply ACLs only to inbound management traffic on Layer 3 interfaces.

Follow these steps to control access to an interface.

SUMMARY STEPS

1. enable
2. configure terminal
3. interface *interface-id*
4. no switchport
5. ipv6 address *ipv6-address*
6. ipv6 traffic-filter *access-list-name* {in | out}
7. end
8. show running-config
9. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Identify a Layer 2 interface (for port ACLs) or Layer 3 interface (for router ACLs) on which to apply an access list, and enter interface configuration mode.
Step 4	no switchport	If applying a router ACL, this changes the interface from Layer 2 mode (the default) to Layer 3 mode.
Step 5	ipv6 address <i>ipv6-address</i>	Configure an IPv6 address on a Layer 3 interface (for router ACLs).
Step 6	ipv6 traffic-filter <i>access-list-name</i> { in out }	Apply the access list to incoming or outgoing traffic on the interface. Note The out keyword is not supported for Layer 2 interfaces (port ACLs).
Step 7	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Switch# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring IPv6 ACLs

You can display information about all configured access lists, all IPv6 access lists, or a specific access list by using one or more of the privileged EXEC commands shown in the table below:

Table 134: show ACL commands

Command	Purpose
show access-lists	Displays all access lists configured on the switch.
show ipv6 access-list [<i>access-list-name</i>]	Displays all configured IPv6 access lists or the access list specified by name.
show vlan access-map [<i>map-name</i>]	Displays VLAN access map configuration.
show vlan filter [access-map <i>access-map</i> vlan <i>vlan-id</i>]	Displays the mapping between VACLs and VLANs.

This is an example of the output from the `show access-lists` privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Switch # show access-lists
Extended IP access list hello
  10 permit ip any any
IPv6 access list ipv6
  permit ipv6 any any sequence 10
```

This is an example of the output from the `show ipv6 access-list` privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack

```
Switch# show ipv6 access-list
IPv6 access list inbound
  permit tcp any any eq bgp (8 matches) sequence 10
  permit tcp any any eq telnet (15 matches) sequence 20
  permit udp any any sequence 30
IPv6 access list outbound
  deny udp any any sequence 10
  deny tcp any any eq telnet sequence 20
```

This is an example of the output from the `show vlan access-map` privileged EXEC command. The output shows VLAN access map information.

```
Switch# show vlan access-map
Vlan access-map "m1" 10
  Match clauses:
    ipv6 address: ip2
  Action: drop
```




CHAPTER 65

Configuring DHCP

- [Restrictions for DHCP, on page 1277](#)
- [Information About DHCP, on page 1277](#)
- [How to Configure DHCP Features, on page 1284](#)
- [Configuring DHCP Server Port-Based Address Allocation, on page 1293](#)

Restrictions for DHCP

The following scenario is not supported:

A non-DHCP snooping VLAN, and the SVI of the non-DHCP snooping VLAN is configured on a device. The SVI of the non-DHCP snooping VLAN is configured with the status of *no shutdown*. In this scenario, the DHCP packets in the non-DHCP snooping VLAN are not forwarded to the trusted ports.

If the SVI of the non-DHCP snooping VLAN is not configured or is configured with the *shutdown* status, DHCP packets are forwarded to the trusted ports, and DHCP clients can obtain IP address from the DHCP server.

Information About DHCP

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it forwards the request to one or more secondary DHCP servers defined by the network administrator. The switch can act as a DHCP server.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on output interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.



Note For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces, as untrusted DHCP messages will be forwarded only to trusted interfaces.

An untrusted DHCP message is a message that is received through an untrusted interface. By default, the switch considers all interfaces untrusted. So, the switch must be configured to trust some interfaces to use DHCP Snooping. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer's switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database has the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not have information regarding hosts interconnected with a trusted interface.



Note When configuring DHCP snooping to block unauthorized IP address using the **ip verify source prot-security** command on an interface, the **switchport port-security** command should also be configured.

In a service-provider network, an example of an interface you might configure as trusted is one connected to a port on a device in the same network. An example of an untrusted interface is one that is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If the addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCP OFFER, DHCP ACK, DHCP NAK, or DHCP REQUEST packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCP RELEASE or DHCP DECLINE broadcast message that has a MAC address in the DHCP snooping binding database, but the interface information in the binding database does not match the interface on which the message was received.
- A DHCP relay agent forwards a DHCP packet that includes a relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allow-untrusted** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP inspection or IP source guard, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on untrusted input interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

Normally, it is not desirable to broadcast packets to wireless clients. So, DHCP snooping replaces destination broadcast MAC address (ffff.ffff.ffff) with unicast MAC address for DHCP packets that are going from server to wireless clients. The unicast MAC address is retrieved from CHADDR field in the DHCP payload. This processing is applied for server to client packets such as DHCP OFFER, DHCP ACK, and DHCP NACK messages. The **ip dhcp snooping wireless bootp-broadcast enable** can be used to revert this behavior. When the wireless BOOTP broadcast is enabled, the broadcast DHCP packets from server are forwarded to wireless clients without changing the destination MAC address.

Option-82 Data Insertion

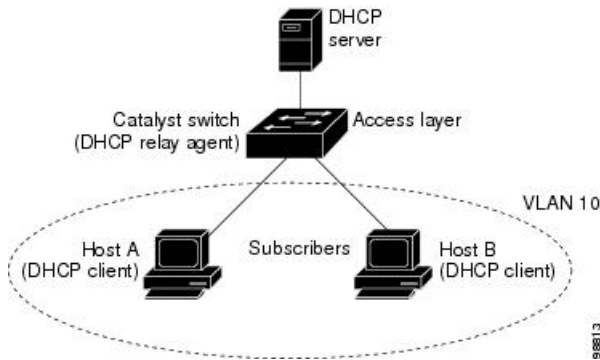
In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.



Note The DHCP option-82 feature is supported only when DHCP snooping is globally enabled on the VLANs to which subscriber devices using option-82 are assigned.

The following illustration shows a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Catalyst switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 103: DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option 82 on the switch, the following sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.
- When the switch receives the DHCP request, it adds the option-82 information in the packet. By default, the remote-ID suboption is the switch MAC address, and the circuit-ID suboption is the port identifier, **vlan-mod-port**, from which the packet is received. You can configure the remote ID and circuit ID.
- If the IP address of the relay agent is configured, the switch adds this IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82-capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. Then the DHCP server echoes the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.

In the default suboption configuration, when the described sequence of events occurs, the values in these fields do not change (see the illustration, *Suboption Packet Formats*):

- Circuit-ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Circuit-ID type
 - Length of the circuit-ID type
- Remote-ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Remote-ID type

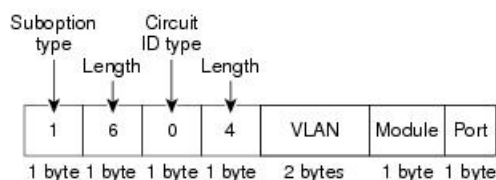
- Length of the remote-ID type

In the port field of the circuit ID suboption, the port numbers start at 3. For example, on a switch with 24 10/100/1000 ports and four small form-factor pluggable (SFP) module slots, port 3 is the Gigabit Ethernet 1/0/1 port, port 4 is the Gigabit Ethernet 1/0/2 port, and so forth. Port 27 is the SFP module slot Gigabit Ethernet1/0/25, and so forth.

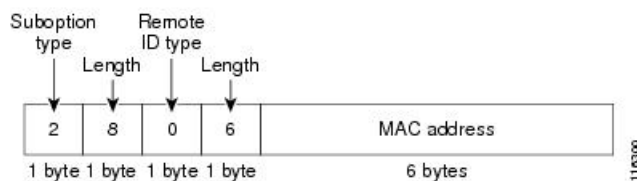
The illustration, *Suboption Packet Formats*, shows the packet formats for the remote-ID suboption and the circuit-ID suboption when the default suboption configuration is used. For the circuit-ID suboption, the module number corresponds to the switch number in the stack. The switch uses the packet formats when you globally enable DHCP snooping and enter the `ip dhcp snooping information option global` configuration command.

Figure 104: Suboption Packet Formats

Circuit ID Suboption Frame Format



Remote ID Suboption Frame Format

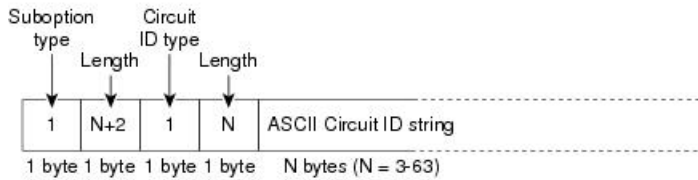
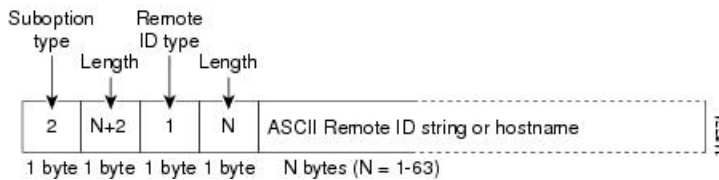


The illustration, *User-Configured Suboption Packet Formats*, shows the packet formats for user-configured remote-ID and circuit-ID suboptions. The switch uses these packet formats when DHCP snooping is globally enabled and when the `ip dhcp snooping information option format remote-id` global configuration command and the `ip dhcp snooping vlan information option format-type circuit-id string` interface configuration command are entered.

The values for these fields in the packets change from the default values when you configure the remote-ID and circuit-ID suboptions:

- Circuit-ID suboption fields
 - The circuit-ID type is 1.
 - The length values are variable, depending on the length of the string that you configure.
- Remote-ID suboption fields
 - The remote-ID type is 1.
 - The length values are variable, depending on the length of the string that you configure.

Figure 105: User-Configured Suboption Packet Formats

Circuit ID Suboption Frame Format (for user-configured string):**Remote ID Suboption Frame Format (for user-configured string):**

Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, address bindings, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool. For more information about manual and automatic address bindings, see the “Configuring DHCP” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

For procedures to enable and configure the Cisco IOS DHCP server database, see the “DHCP Configuration Task List” section in the “Configuring DHCP” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 64,000 bindings.

Each database entry (binding) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. The database agent stores the bindings in a file at a configured location. At the end of each entry is a checksum that accounts for all the bytes from the start of the file through all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP inspection or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DHCP spoofing attacks.

When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch updates the file when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is

updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and cancel-timeout values), the update stops.

This is the format of the file with bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
<entry-n> <checksum-1-2-...-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The initial-checksum entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Gi1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Gi1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Gi1/0/4 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

How to Configure DHCP Features

Default DHCP Snooping Configuration

Table 135: Default DHCP Configuration

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration ¹³
DHCP relay agent	Enabled ¹⁴
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)
DHCP relay agent forwarding policy	Replace the existing relay agent information
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted input interfaces ¹⁵	Disabled
DHCP snooping limit rate	None configured
DHCP snooping trust	Untrusted
DHCP snooping VLAN	Disabled
DHCP snooping MAC address verification	Enabled
Cisco IOS DHCP server binding database	Enabled in Cisco IOS software, requires configuration. Note The switch gets network addresses and configuration parameters only from a device configured as a DHCP server.
DHCP snooping binding database agent	Enabled in Cisco IOS software, requires configuration. This feature is operational only when a destination is configured.

¹³ The switch responds to DHCP requests only if it is configured as a DHCP server.

¹⁴ The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.

¹⁵ Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.

DHCP Snooping Configuration Guidelines

- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust interface** configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust interface** configuration command.
- You can display DHCP snooping statistics by entering the **show ip dhcp snooping statistics** user EXEC command, and you can clear the snooping statistics counters by entering the **clear ip dhcp snooping statistics** privileged EXEC command.

Configuring the DHCP Server

The switch can act as a DHCP server.

For procedures to configure the switch as a DHCP server, see the “Configuring DHCP” section of the “IP addressing and Services” section of the *Cisco IOS IP Configuration Guide, Release 12.4*.

Configuring the DHCP Relay Agent

Follow these steps to enable the DHCP relay agent on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **service dhcp**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	service dhcp Example:	Enables the DHCP server and relay agent on your switch. By default, this feature is enabled.

	Command or Action	Purpose
	Switch(config)# service dhcp	
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

- Checking (validating) the relay agent information
- Configuring the relay agent forwarding policy

Specifying the Packet Forwarding Address

If the DHCP server and the DHCP clients are on different networks or subnets, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

Beginning in privileged EXEC mode, follow these steps to specify the packet forwarding address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface vlan** *vlan-id*
4. **ip address** *ip-address subnet-mask*
5. **ip helper-address** *address*
6. **end**
7. Use one of the following:
 - **interface range** *port-range*
 - **interface** *interface-id*

8. `switchport mode access`
9. `switchport access vlan vlan-id`
10. `end`
11. `show running-config`
12. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>interface vlan <i>vlan-id</i></code></p> <p>Example:</p> <pre>Switch(config)# interface vlan 1</pre>	<p>Creates a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.</p>
Step 4	<p><code>ip address <i>ip-address subnet-mask</i></code></p> <p>Example:</p> <pre>Switch(config-if)# ip address 192.108.1.27 255.255.255.0</pre>	<p>Configures the interface with an IP address and an IP subnet.</p>
Step 5	<p><code>ip helper-address <i>address</i></code></p> <p>Example:</p> <pre>Switch(config-if)# ip helper-address 172.16.1.2</pre>	<p>Specifies the DHCP packet forwarding address.</p> <p>The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests.</p> <p>If you have multiple servers, you can configure one helper address for each server.</p>
Step 6	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	<p>Returns to global configuration mode.</p>
Step 7	<p>Use one of the following:</p> <ul style="list-style-type: none"> • <code>interface range <i>port-range</i></code> • <code>interface <i>interface-id</i></code> 	<p>Configures multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode.</p>

	Command or Action	Purpose
	Example: <pre>Switch(config)# interface gigabitethernet1/0/2</pre>	or Configures a single physical port that is connected to the DHCP client, and enter interface configuration mode.
Step 8	switchport mode access Example: <pre>Switch(config-if)# switchport mode access</pre>	Defines the VLAN membership mode for the port.
Step 9	switchport access vlan <i>vlan-id</i> Example: <pre>Switch(config-if)# switchport access vlan 1</pre>	Assigns the ports to the same VLAN as configured in Step 2.
Step 10	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 11	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 12	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Prerequisites for Configuring DHCP Snooping and Option 82

The prerequisites for DHCP Snooping and Option 82 are as follows:

- You must globally enable DHCP snooping on the switch.
- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- If you want the switch to respond to DHCP requests, it must be configured as a DHCP server.
- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces. In a service-provider network, a trusted interface is connected to a port on a device in the same network.

- You must configure the switch to use the Cisco IOS DHCP server binding database to use it for DHCP snooping.
- To use the DHCP snooping option of accepting packets on untrusted inputs, the switch must be an aggregation switch that receives packets with option-82 information from an edge switch.
- The following prerequisites apply to DHCP snooping binding database configuration:
 - You must configure a destination on the DHCP snooping binding database to use the switch for DHCP snooping.
 - Because both NVRAM and the flash memory have limited storage capacity, we recommend that you store the binding file on a TFTP server.
 - For network-based URLs (such as TFTP and FTP), you must create an empty file at the configured URL before the switch can write bindings to the binding file at that URL. See the documentation for your TFTP server to determine whether you must first create an empty file on the server; some TFTP servers cannot be configured this way.
 - To ensure that the lease time in the database is accurate, we recommend that you enable and configure Network Time Protocol (NTP).
 - If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- If you want the switch to relay DHCP packets, the IP address of the DHCP server must be configured on the switch virtual interface (SVI) of the DHCP client.
- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust interface** configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.

Enabling DHCP Snooping and Option 82

Follow these steps to enable DHCP snooping on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip dhcp snooping**
4. **ip dhcp snooping vlan** *vlan-range*
5. **ip dhcp snooping information option**
6. **ip dhcp snooping information option format remote-id** [*string ASCII-string* | **hostname**]
7. **ip dhcp snooping information option allow-untrusted**
8. **interface** *interface-id*
9. **ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] *string ASCII-string*

10. `ip dhcp snooping trust`
11. `ip dhcp snooping limit rate rate`
12. `exit`
13. `ip dhcp snooping verify mac-address`
14. `end`
15. `show running-config`
16. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip dhcp snooping Example: <pre>Switch(config)# ip dhcp snooping</pre>	Enables DHCP snooping globally.
Step 4	ip dhcp snooping vlan <i>vlan-range</i> Example: <pre>Switch(config)# ip dhcp snooping vlan 10</pre>	Enables DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4094. You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space. <ul style="list-style-type: none"> • You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
Step 5	ip dhcp snooping information option Example: <pre>Switch(config)# ip dhcp snooping information option</pre>	Enables the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. This is the default setting.
Step 6	ip dhcp snooping information option format remote-id [string <i>ASCII-string</i> hostname]	(Optional) Configures the remote-ID suboption. You can configure the remote ID as:

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(config)# ip dhcp snooping information option format remote-id string acsiistring2</pre>	<ul style="list-style-type: none"> String of up to 63 ASCII characters (no spaces) Configured hostname for the switch <p>Note If the hostname is longer than 63 characters, it is truncated to 63 characters in the remote-ID configuration.</p> <p>The default remote ID is the switch MAC address.</p>
Step 7	<p>ip dhcp snooping information option allow-untrusted</p> <p>Example:</p> <pre>Switch(config)# ip dhcp snooping information option allow-untrusted</pre>	<p>(Optional) If the switch is an aggregation switch connected to an edge switch, this command enables the switch to accept incoming DHCP snooping packets with option-82 information from the edge switch.</p> <p>The default setting is disabled.</p> <p>Note Enter this command only on aggregation switches that are connected to trusted devices.</p>
Step 8	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	<p>Specifies the interface to be configured, and enter interface configuration mode.</p>
Step 9	<p>ip dhcp snooping vlan <i>vlan</i> information option format-type circuit-id [override] string <i>ASCII-string</i></p> <p>Example:</p> <pre>Switch(config-if)# ip dhcp snooping vlan 1 information option format-type circuit-id override string override2</pre>	<p>(Optional) Configures the circuit-ID suboption for the specified interface.</p> <p>Specify the VLAN and port identifier, using a VLAN ID in the range of 1 to 4094. The default circuit ID is the port identifier, in the format vlan-mod-port.</p> <p>You can configure the circuit ID to be a string of 3 to 63 ASCII characters (no spaces).</p> <p>(Optional) Use the override keyword when you do not want the circuit-ID suboption inserted in TLV format to define subscriber information.</p>
Step 10	<p>ip dhcp snooping trust</p> <p>Example:</p> <pre>Switch(config-if)# ip dhcp snooping trust</pre>	<p>(Optional) Configures the interface as trusted or untrusted. Use the no keyword to configure an interface to receive messages from an untrusted client. The default setting is untrusted.</p>
Step 11	<p>ip dhcp snooping limit rate <i>rate</i></p> <p>Example:</p> <pre>Switch(config-if)# ip dhcp snooping limit rate 100</pre>	<p>(Optional) Configures the number of DHCP packets per second that an interface can receive. The range is 1 to 2048. By default, no rate limit is configured.</p>

	Command or Action	Purpose
		Note We recommend an untrusted rate limit of not more than 100 packets per second. If you configure rate limiting for trusted interfaces, you might need to increase the rate limit if the port is a trunk port assigned to more than one VLAN with DHCP snooping.
Step 12	exit Example: Switch(config-if)# exit	Returns to global configuration mode.
Step 13	ip dhcp snooping verify mac-address Example: Switch(config)# ip dhcp snooping verify mac-address	(Optional) Configures the switch to verify that the source MAC address in a DHCP packet received on untrusted ports matches the client hardware address in the packet. The default is to verify that the source MAC address matches the client hardware address in the packet.
Step 14	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 15	show running-config Example: Switch# show running-config	Verifies your entries.
Step 16	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling the Cisco IOS DHCP Server Database

For procedures to enable and configure the Cisco IOS DHCP server database, see the “DHCP Configuration Task List” section in the “Configuring DHCP” chapter of the Cisco IOS IP Configuration Guide, Release 12.4

Monitoring DHCP Snooping Information

Table 136: Commands for Displaying DHCP Information

show ip dhcp snooping	Displays the DHCP snooping configuration for a switch
------------------------------	-------------------------------------------------------

show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding table, also referred to as a binding table.
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip dhcp snooping statistics	Displays the DHCP snooping statistics in summary or detail form.
show ip source binding	Display the dynamically and statically configured bindings.



Note If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the statically configured bindings.

Configuring DHCP Server Port-Based Address Allocation

Information About Configuring DHCP Server Port-Based Address Allocation

DHCP server port-based address allocation is a feature that enables DHCP to maintain the same IP address on an Ethernet switch port regardless of the attached device client identifier or client hardware address.

When Ethernet switches are deployed in the network, they offer connectivity to the directly connected devices. In some environments, such as on a factory floor, if a device fails, the replacement device must be working immediately in the existing network. With the current DHCP implementation, there is no guarantee that DHCP would offer the same IP address to the replacement device. Control, monitoring, and other software expect a stable IP address associated with each device. If a device is replaced, the address assignment should remain stable even though the DHCP client has changed.

When configured, the DHCP server port-based address allocation feature ensures that the same IP address is always offered to the same connected port even as the client identifier or client hardware address changes in the DHCP messages received on that port. The DHCP protocol recognizes DHCP clients by the client identifier option in the DHCP packet. Clients that do not include the client identifier option are identified by the client hardware address. When you configure this feature, the port name of the interface overrides the client identifier or hardware address and the actual point of connection, the switch port, becomes the client identifier.

In all cases, by connecting the Ethernet cable to the same port, the same IP address is allocated through DHCP to the attached device.

The DHCP server port-based address allocation feature is only supported on a Cisco IOS DHCP server and not a third-party server.

Default Port-Based Address Allocation Configuration

By default, DHCP server port-based address allocation is disabled.

Port-Based Address Allocation Configuration Guidelines

- By default, DHCP server port-based address allocation is disabled.

- To restrict assignments from the DHCP pool to preconfigured reservations (unreserved addresses are not offered to the client and other clients are not served by the pool), you can enter the **reserved-only** DHCP pool configuration command.

Enabling the DHCP Snooping Binding Database Agent

Beginning in privileged EXEC mode, follow these steps to enable and configure the DHCP snooping binding database agent on the switch:

SUMMARY STEPS

- enable**
- configure terminal**
- ip dhcp snooping database** {flash[number]:/filename | ftp://user:password@host/filename | http://[[username:password]@]{hostname | host-ip}{/directory] /image-name.tar | rcp://user@host/filename} | tftp://host/filename
- ip dhcp snooping database timeout** seconds
- ip dhcp snooping database write-delay** seconds
- end**
- ip dhcp snooping binding** mac-address **vlan** vlan-id ip-address **interface** interface-id **expiry** seconds
- show ip dhcp snooping database** [detail]
- show running-config**
- copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip dhcp snooping database {flash[number]:/filename ftp://user:password@host/filename http://[[username:password]@]{hostname host-ip}{/directory] /image-name.tar rcp://user@host/filename} tftp://host/filename Example: Switch(config)# ip dhcp snooping database tftp://10.90.90.90/snooping-rp2	Specifies the URL for the database agent or the binding file by using one of these forms: <ul style="list-style-type: none"> flash[number]:/filename (Optional) Use the <i>number</i> parameter to specify the stack member number of the active switch. The range for <i>number</i> is 1 to 9. ftp://user:password@host/filename

	Command or Action	Purpose
		<ul style="list-style-type: none"> • http://[<i>username:password</i>]@]{<i>hostname / host-ip</i>}[<i>/directory</i>] <i>/image-name.tar</i> • rcp://<i>user@host/filename</i> • tftp://<i>host/filename</i>
Step 4	ip dhcp snooping database timeout <i>seconds</i> Example: <pre>Switch(config)# ip dhcp snooping database timeout 300</pre>	Specifies (in seconds) how long to wait for the database transfer process to finish before stopping the process. The default is 300 seconds. The range is 0 to 86400. Use 0 to define an infinite duration, which means to continue trying the transfer indefinitely.
Step 5	ip dhcp snooping database write-delay <i>seconds</i> Example: <pre>Switch(config)# ip dhcp snooping database write-delay 15</pre>	Specifies the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	ip dhcp snooping binding <i>mac-address</i> vlan <i>vlan-id</i> <i>ip-address</i> interface <i>interface-id</i> expiry <i>seconds</i> Example: <pre>Switch# ip dhcp snooping binding 0001.1234.1234 vlan 1 172.20.50.5 interface gil/1 expiry 1000</pre>	(Optional) Adds binding entries to the DHCP snooping binding database. The <i>vlan-id</i> range is from 1 to 4904. The <i>seconds</i> range is from 1 to 4294967295. Enter this command for each entry that you add. Use this command when you are testing or debugging the switch.
Step 8	show ip dhcp snooping database [detail] Example: <pre>Switch# show ip dhcp snooping database detail</pre>	Displays the status and statistics of the DHCP snooping binding database agent.
Step 9	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 10	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling DHCP Server Port-Based Address Allocation

Follow these steps to globally enable port-based address allocation and to automatically generate a subscriber identifier on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip dhcp use subscriber-id client-id**
4. **ip dhcp subscriber-id interface-name**
5. **interface *interface-id***
6. **ip dhcp server use subscriber-id client-id**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip dhcp use subscriber-id client-id Example: Switch(config)# ip dhcp use subscriber-id client-id	Configures the DHCP server to globally use the subscriber identifier as the client identifier on all incoming DHCP messages.
Step 4	ip dhcp subscriber-id interface-name Example: Switch(config)# ip dhcp subscriber-id interface-name	Automatically generates a subscriber identifier based on the short name of the interface. A subscriber identifier configured on a specific interface takes precedence over this command.
Step 5	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Specifies the interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 6	ip dhcp server use subscriber-id client-id Example: <pre>Switch(config-if)# ip dhcp server use subscriber-id client-id</pre>	Configures the DHCP server to use the subscriber identifier as the client identifier on all incoming DHCP messages on the interface.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

After enabling DHCP port-based address allocation on the switch, use the **ip dhcp pool** global configuration command to preassign IP addresses and to associate them to clients.

Monitoring DHCP Server Port-Based Address Allocation

Table 137: Commands for Displaying DHCP Port-Based Address Allocation Information

Command	Purpose
show interface <i>interface id</i>	Displays the status and configuration of a specific interface.
show ip dhcp pool	Displays the DHCP address pools.
show ip dhcp binding	Displays address bindings on the Cisco IOS DHCP server.



CHAPTER 66

Configuring IP Source Guard

IP Source Guard (IPSG) is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured IP source bindings.

This chapter contains the following topics:

- [Information About IP Source Guard, on page 1299](#)
- [How to Configure IP Source Guard, on page 1301](#)
- [Monitoring IP Source Guard, on page 1304](#)

Information About IP Source Guard

IP Source Guard

You can use IP source guard to prevent traffic attacks if a host tries to use the IP address of its neighbor and you can enable IP source guard when DHCP snooping is enabled on an untrusted interface.

After IPSG is enabled on an interface, the switch blocks all IP traffic received on the interface except for DHCP packets allowed by DHCP snooping.

The switch uses a source IP lookup table in hardware to bind IP addresses to ports. For IP and MAC filtering, a combination of source IP and source MAC lookups are used. IP traffic with a source IP address in the binding table is allowed, all other traffic is denied.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IPSG is supported only on Layer 2 ports, including access and trunk ports. You can configure IPSG with source IP address filtering or with source IP and MAC address filtering.

IP Source Guard for Static Hosts



Note Do not use IPSG (IP source guard) for static hosts on uplink ports or trunk ports.

IPSG for static hosts extends the IPSG capability to non-DHCP and static environments. The previous IPSG used the entries created by DHCP snooping to validate the hosts connected to a switch. Any traffic received from a host without a valid DHCP binding entry is dropped. This security feature restricts IP traffic on nonrouted Layer 2 interfaces. It filters traffic based on the DHCP snooping binding database and on manually configured IP source bindings. The previous version of IPSG required a DHCP environment for IPSG to work.

IPSG for static hosts allows IPSG to work without DHCP. IPSG for static hosts relies on IP device tracking-table entries to install port ACLs. The switch creates static entries based on ARP requests or other IP packets to maintain the list of valid hosts for a given port. You can also specify the number of hosts allowed to send traffic to a given port. This is equivalent to port security at Layer 3.

IPSG for static hosts also supports dynamic hosts. If a dynamic host receives a DHCP-assigned IP address that is available in the IP DHCP snooping table, the same entry is learned by the IP device tracking table. In a stacked environment, when the active switch failover occurs, the IP source guard entries for static hosts attached to member ports are retained. When you enter the **show ip device tracking all EXEC** command, the IP device tracking table displays the entries as ACTIVE.



Note Some IP hosts with multiple network interfaces can inject some invalid packets into a network interface. The invalid packets contain the IP or MAC address for another network interface of the host as the source address. The invalid packets can cause IPSG for static hosts to connect to the host, to learn the invalid IP or MAC address bindings, and to reject the valid bindings. Consult the vendor of the corresponding operating system and the network interface to prevent the host from injecting invalid packets.

IPSG for static hosts initially learns IP or MAC bindings dynamically through an ACL-based snooping mechanism. IP or MAC bindings are learned from static hosts by ARP and IP packets. They are stored in the device tracking database. When the number of IP addresses that have been dynamically learned or statically configured on a given port reaches a maximum, the hardware drops any packet with a new IP address. To resolve hosts that have moved or gone away for any reason, IPSG for static hosts leverages IP device tracking to age out dynamically learned IP address bindings. This feature can be used with DHCP snooping. Multiple bindings are established on a port that is connected to both DHCP and static hosts. For example, bindings are stored in both the device tracking database as well as in the DHCP snooping binding database.

IP Source Guard Configuration Guidelines

- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding mac-address vlan vlan-id ip-address interface interface-id** global configuration command on a routed interface, this error message appears:

```
Static IP source binding can only be configured on switch port.
```

- When IP source guard with source IP filtering is enabled on an interface, DHCP snooping must be enabled on the access VLAN for that interface.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.



Note If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.

- You can enable this feature when 802.1x port-based authentication is enabled.
- When you configure IP source guard smart logging, packets with a source address other than the specified address or an address learned by DHCP are denied, and the packet contents are sent to a NetFlow collector. If you configure this feature, make sure that smart logging is globally enabled.
- In a switch stack, if IP source guard is configured on a stack member interface and you remove the configuration of that switch by entering the **no switch stack-member-number provision** global configuration command, the interface static bindings are removed from the binding table, but they are not removed from the running configuration. If you again provision the switch by entering the **switch stack-member-number provision** command, the binding is restored.

To remove the binding from the running configuration, you must disable IP source guard before entering the **no switch provision** command. The configuration is also removed if the switch reloads while the interface is removed from the binding table.

How to Configure IP Source Guard

Enabling IP Source Guard

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip verify source** [**mac-check**]
5. **exit**
6. **ip source binding** *mac-address* **vlan** *vlan-id* *ip-address* **interface** *interface-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip verify source [mac-check] Example: Switch(config-if)# ip verify source	Enables IP source guard with source IP address filtering. (Optional) mac-check —Enables IP Source Guard with source IP address and MAC address filtering.
Step 5	exit Example: Switch(config-if)# exit	Returns to global configuration mode.
Step 6	ip source binding <i>mac-address</i> <i>vlan</i> <i>vlan-id</i> <i>ip-address</i> interface <i>interface-id</i> Example: Switch(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1	Adds a static IP source binding. Enter this command for each static binding.
Step 7	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Switch# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port

You must configure the **ip device tracking maximum *limit-number*** interface configuration command globally for IPSG for static hosts to work. If you only configure this command on a port without enabling IP device tracking globally or by setting an IP device tracking maximum on that interface, IPSG with static hosts rejects all the IP traffic from that interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip device tracking**
4. **interface *interface-id***
5. **switchport mode access**
6. **switchport access vlan *vlan-id***
7. **ip verify source[tracking] [mac-check]**
8. **ip device tracking maximum *number***
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip device tracking Example: <pre>Switch(config)# ip device tracking</pre>	Turns on the IP host table, and globally enables IP device tracking.
Step 4	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode.
Step 5	switchport mode access Example:	Configures a port as access.

	Command or Action	Purpose
	Switch(config-if) # <code>switchport mode access</code>	
Step 6	switchport access vlan <i>vlan-id</i> Example: Switch(config-if) # <code>switchport access vlan 10</code>	Configures the VLAN for this port.
Step 7	ip verify source [tracking] [mac-check] Example: Switch(config-if) # <code>ip verify source tracking mac-check</code>	Enables IP source guard with source IP address filtering. (Optional) tracking —Enables IP source guard for static hosts. (Optional) mac-check —Enables MAC address filtering. The command ip verify source tracking mac-check enables IP source guard for static hosts with MAC address filtering.
Step 8	ip device tracking maximum <i>number</i> Example: Switch(config-if) # <code>ip device tracking maximum 8</code>	Establishes a maximum limit for the number of static IPs that the IP device tracking table allows on the port. The range is 1 to 10. The maximum number is 10. Note You must configure the ip device tracking maximum limit-number interface configuration command.
Step 9	end Example: Switch(config) # <code>end</code>	Returns to privileged EXEC mode.

Monitoring IP Source Guard

Table 138: Privileged EXEC show Commands

Command	Purpose
<code>show ip verify source [interface interface-id]</code>	Displays the IP source guard configuration on the switch or on a specific interface.
<code>show ip device tracking { all interface interface-id ip ip-address mac mac-address }</code>	Displays information about the entries in the IP device tracking table.

Table 139: Interface Configuration Commands

Command	Purpose
ip verify source tracking	Verifies the data source.

For detailed information about the fields in these displays, see the command reference for this release.



CHAPTER 67

Configuring Dynamic ARP Inspection

- [Restrictions for Dynamic ARP Inspection, on page 1307](#)
- [Understanding Dynamic ARP Inspection, on page 1308](#)
- [Default Dynamic ARP Inspection Configuration, on page 1312](#)
- [Relative Priority of ARP ACLs and DHCP Snooping Entries, on page 1312](#)
- [Configuring ARP ACLs for Non-DHCP Environments, on page 1312](#)
- [Configuring Dynamic ARP Inspection in DHCP Environments, on page 1315](#)
- [Limiting the Rate of Incoming ARP Packets, on page 1317](#)
- [Performing Dynamic ARP Inspection Validation Checks, on page 1319](#)
- [Monitoring DAI, on page 1321](#)
- [Verifying the DAI Configuration, on page 1321](#)

Restrictions for Dynamic ARP Inspection

This section lists the restrictions and guidelines for configuring Dynamic ARP Inspection on the switch.

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.
- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.
- Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.

- Dynamic ARP inspection is supported on access ports, trunk ports, and EtherChannel ports.



Note Do not enable Dynamic ARP inspection on RSPAN VLANs. If Dynamic ARP inspection is enabled on RSPAN VLANs, Dynamic ARP inspection packets might not reach the RSPAN destination port.

- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.

- The rate limit is calculated separately on each switch in a switch stack. For a cross-stack EtherChannel, this means that the actual rate limit might be higher than the configured value. For example, if you set the rate limit to 30 pps on an EtherChannel that has one port on switch 1 and one port on switch 2, each port can receive packets at 29 pps without causing the EtherChannel to become error-disabled.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.

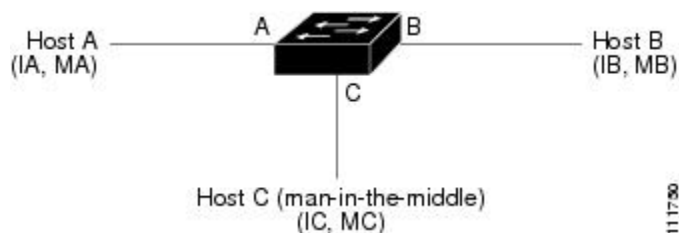
- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple dynamic ARP inspection-enabled VLANs. You also can use the **ip arp inspection limit none** interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.
- When you enable dynamic ARP inspection on the switch, policers that were configured to police ARP traffic are no longer effective. The result is that all ARP traffic is sent to the CPU.

Understanding Dynamic ARP Inspection

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However, because ARP allows a gratuitous reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 26-1 shows an example of ARP cache poisoning.

Figure 106: ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the-middle* attack.

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports
- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

Dynamic ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan *vlan-range*** global configuration command.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list *acl-name*** global configuration command.

You can configure dynamic ARP inspection to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header. Use the **ip arp inspection validate {[*src-mac*] [*dst-mac*] [*ip*]}** global configuration command.

Interface Trust States and Network Security

Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

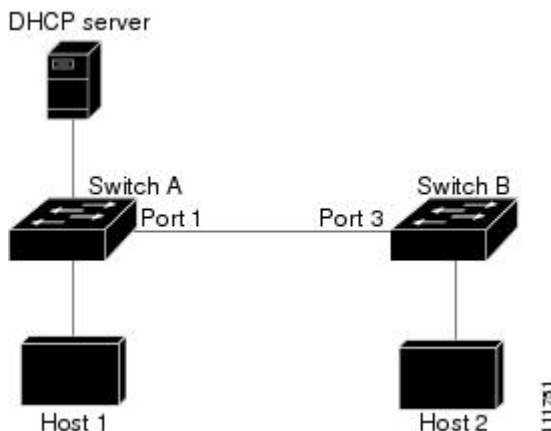
In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using their `arp inspection trust interface` configuration command.



Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In the following figure, assume that both Switch A and Switch B are running dynamic ARP inspection on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 107: ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection switches.



Note Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using the **ip arp inspection limit** interface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the **errdisable recovery** global configuration command to enable error disable recovery so that ports automatically emerge from this state after a specified timeout period.



Note The rate limit for an EtherChannel is applied separately to each switch in a stack. For example, if a limit of 20 pps is configured on the EtherChannel, each switch with ports in the EtherChannel can carry up to 20 pps. If any switch exceeds the limit, the entire EtherChannel is placed into the error-disabled state.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command.

Default Dynamic ARP Inspection Configuration

Feature	Default Settings
Dynamic ARP inspection	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Rate limit of incoming ARP packets	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
ARP ACLs for non-DHCP environments	No ARP ACLs are defined.
Validation checks	No checks are performed.
Log buffer	When dynamic ARP inspection is enabled, all denied or dropped ARP packets are logged. The number of entries in the log is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second.
Per-VLAN logging	All denied or dropped ARP packets are logged.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the `ip arp inspection filter vlan` global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 2 does not support dynamic ARP inspection or DHCP snooping.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

Follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **arp access-list *acl-name***
4. **permit ip host *sender-ip* mac host *sender-mac***
5. **exit**
6. **ip arp inspection filter *arp-acl-name* vlan *vlan-range* [static]**
7. **interface *interface-id***
8. **no ip arp inspection trust**
9. **end**
10. Use the following show commands:
 - **show arp access-list *acl-name***
 - **show ip arp inspection vlan *vlan-range***
 - **show ip arp inspection interfaces**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	arp access-list <i>acl-name</i>	Defines an ARP ACL, and enters ARP access-list configuration mode. By default, no ARP access lists are defined. <p>Note At the end of the ARP access list, there is an implicit deny ip any mac any command.</p>
Step 4	permit ip host <i>sender-ip</i> mac host <i>sender-mac</i>	Permits ARP packets from the specified host (Host 2). <ul style="list-style-type: none"> • For <i>sender-ip</i>, enter the IP address of Host 2. • For <i>sender-mac</i>, enter the MAC address of Host 2.

	Command or Action	Purpose
Step 5	<code>exit</code>	Returns to global configuration mode.
Step 6	<code>ip arp inspection filter <i>arp-acl-name</i> vlan <i>vlan-range</i> [static]</code>	<p>Applies ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN.</p> <ul style="list-style-type: none"> For <i>arp-acl-name</i>, specify the name of the ACL created in Step 2. For <i>vlan-range</i>, specify the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. (Optional) Specify static to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. <p>If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL.</p> <p>ARP packets containing only IP-to-MAC address bindings are compared against the ACL. Packets are permitted only if the access list permits them.</p>
Step 7	<code>interface <i>interface-id</i></code>	Specifies Switch A interface that is connected to Switch B, and enters the interface configuration mode.
Step 8	<code>no ip arp inspection trust</code>	<p>Configures Switch A interface that is connected to Switch B as untrusted.</p> <p>By default, all interfaces are untrusted.</p> <p>For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the ip arp inspection vlan logging global configuration command.</p>
Step 9	<code>end</code>	Returns to privileged EXEC mode.
Step 10	<p>Use the following show commands:</p> <ul style="list-style-type: none"> <code>show arp access-list <i>acl-name</i></code> <code>show ip arp inspection vlan <i>vlan-range</i></code> <code>show ip arp inspection interfaces</code> 	Verifies your entries.

	Command or Action	Purpose
Step 11	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 12	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Dynamic ARP Inspection in DHCP Environments

Before you begin

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B. Both switches are running dynamic ARP inspection on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.



Note Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

Follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **show cdp neighbors**
3. **configure terminal**
4. **ip arp inspection vlan** *vlan-range*
5. **Interface***interface-id*
6. **ip arp inspection trust**
7. **end**
8. **show ip arp inspection interfaces**
9. **show ip arp inspection vlan** *vlan-range*
10. **show ip dhcp snooping binding**
11. **show ip arp inspection statistics vlan** *vlan-range*
12. **configure terminal**

13. configure terminal

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show cdp neighbors Example: Switch(config-if) # show cdp neighbors	Verify the connection between the switches.
Step 3	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 4	ip arp inspection vlan <i>vlan-range</i> Example: Switch(config) # ip arp inspection vlan 1	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For <i>vlan-range</i> , specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. Specify the same VLAN ID for both switches.
Step 5	Interface <i>interface-id</i> Example: Switch(config) # interface gigabitethernet1/0/1	Specifies the interface connected to the other switch, and enter interface configuration mode.
Step 6	ip arp inspection trust Example: Switch(config-if) # ip arp inspection trust	Configures the connection between the switches as trusted. By default, all interfaces are untrusted. <p>The switch does not check ARP packets that it receives from the other switch on the trusted interface. It simply forwards the packets.</p> <p>For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the <code>ip arp inspection vlan logging global</code> configuration command.</p>
Step 7	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	<code>Switch(config-if)#end</code>	
Step 8	show ip arp inspection interfaces Example:	Verifies the dynamic ARP inspection configuration on interfaces.
Step 9	show ip arp inspection vlan <i>vlan-range</i> Example: <code>Switch(config-if)#show ip arp inspection vlan 1</code>	Verifies the dynamic ARP inspection configuration on VLAN.
Step 10	show ip dhcp snooping binding Example: <code>Switch(config-if)#show ip dhcp snooping binding</code>	Verifies the DHCP bindings.
Step 11	show ip arp inspection statistics vlan <i>vlan-range</i> Example: <code>Switch(config-if)#show ip arp inspection statistics vlan 1</code>	Checks the dynamic ARP inspection statistics on VLAN.
Step 12	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 13	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.

Limiting the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.



Note Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the **no ip arp inspection limit** interface configuration command, the interface reverts to its default rate limit.

Follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip arp inspection limit** {rate pps [burst interval seconds] | none}
5. **exit**
6. Use the following commands:
 - **errdisable detect cause arp-inspection**
 - **errdisable recovery cause arp-inspection**
 - **errdisable recovery interval** *interval*
7. **exit**
8. Use the following show commands:
 - **show ip arp inspection interfaces**
 - **show errdisable recovery**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Specifies the interface to be rate-limited, and enter interface configuration mode.
Step 4	ip arp inspection limit {rate pps [burst interval seconds] none}	Limits the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. The keywords have these meanings: <ul style="list-style-type: none"> • For ratepps, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) For burst interval <i>seconds</i>, specify the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets. The range is 1 to 15. • For rate none, specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 5	exit	Returns to global configuration mode.
Step 6	Use the following commands: <ul style="list-style-type: none"> • errdisable detect cause arp-inspection • errdisable recovery cause arp-inspection • errdisable recovery interval <i>interval</i> 	(Optional) Enables error recovery from the dynamic ARP inspection error-disabled state, and configure the dynamic ARP inspection recover mechanism variables. By default, recovery is disabled, and the recovery interval is 300 seconds. For interval <i>interval</i> , specify the time in seconds to recover from the error-disabled state. The range is 30 to 86400.
Step 7	exit	Returns to privileged EXEC mode.
Step 8	Use the following show commands: <ul style="list-style-type: none"> • show ip arp inspection interfaces • show errdisable recovery 	Verifies your settings.
Step 9	show running-config Example: Switch# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Performing Dynamic ARP Inspection Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address.

Follow these steps to perform specific checks on incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **ip arp inspection validate** {[src-mac] [dst-mac] [ip]}
4. **exit**
5. **show ip arp inspection vlan** *vlan-range*
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip arp inspection validate {[src-mac] [dst-mac] [ip]}</p>	<p>Performs a specific check on incoming ARP packets. By default, no checks are performed.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • For src-mac, check the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For dst-mac, check the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For ip, check the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. <p>You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables src and dst mac validations, and a second command enables IP validation</p>

	Command or Action	Purpose
		only, the src and dst mac validations are disabled as a result of the second command.
Step 4	<code>exit</code>	Returns to privileged EXEC mode.
Step 5	<code>show ip arp inspection vlan <i>vlan-range</i></code>	Verifies your settings.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring DAI

To monitor DAI, use the following commands:

Command	Description
<code>clear ip arp inspection statistics</code>	Clears dynamic ARP inspection statistics.
<code>show ip arp inspection statistics [vlan <i>vlan-range</i>]</code>	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).
<code>clear ip arp inspection log</code>	Clears the dynamic ARP inspection log buffer.
<code>show ip arp inspection log</code>	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For the `show ip arp inspection statistics` command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate.

Verifying the DAI Configuration

To display and verify the DAI configuration, use the following commands:

Command	Description
show arp access-list [<i>acl-name</i>]	Displays detailed information about ARP ACLs.
show ip arp inspection interfaces [interface-id]	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
show ip arp inspection vlan <i>vlan-range</i>	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).



CHAPTER 68

Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication. IEEE 802.1x authentication prevents unauthorized devices (clients) from gaining access to the network. Unless otherwise noted, the term *switch* refers to a standalone switch or a switch stack.

- [Information About 802.1x Port-Based Authentication, on page 1323](#)
- [How to Configure 802.1x Port-Based Authentication, on page 1354](#)
- [Monitoring 802.1x Statistics and Status, on page 1411](#)

Information About 802.1x Port-Based Authentication

The 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.



Note TACACS is not supported with 802.1x authentication.

Until the client is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

Client session	Maximum sessions supported
Maximum dot1x or MAB client sessions	2000
Maximum web-based authentication sessions	2000
Maximum dot1x sessions with critical-auth VLAN enabled and server re-initialized	2000
Maximum MAB sessions with various session features applied	2000

Client session	Maximum sessions supported
Maximum dot1x sessions with service templates or session features applied	2000

Port-Based Authentication Process

To configure IEEE 802.1X port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

The AAA process begins with authentication. When 802.1x port-based authentication is enabled and the client supports 802.1x-compliant client software, these events occur:

- If the client identity is valid and the 802.1x authentication succeeds, the switch grants the client access to the network.
- If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.
- If the switch gets an invalid identity from an 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.
- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

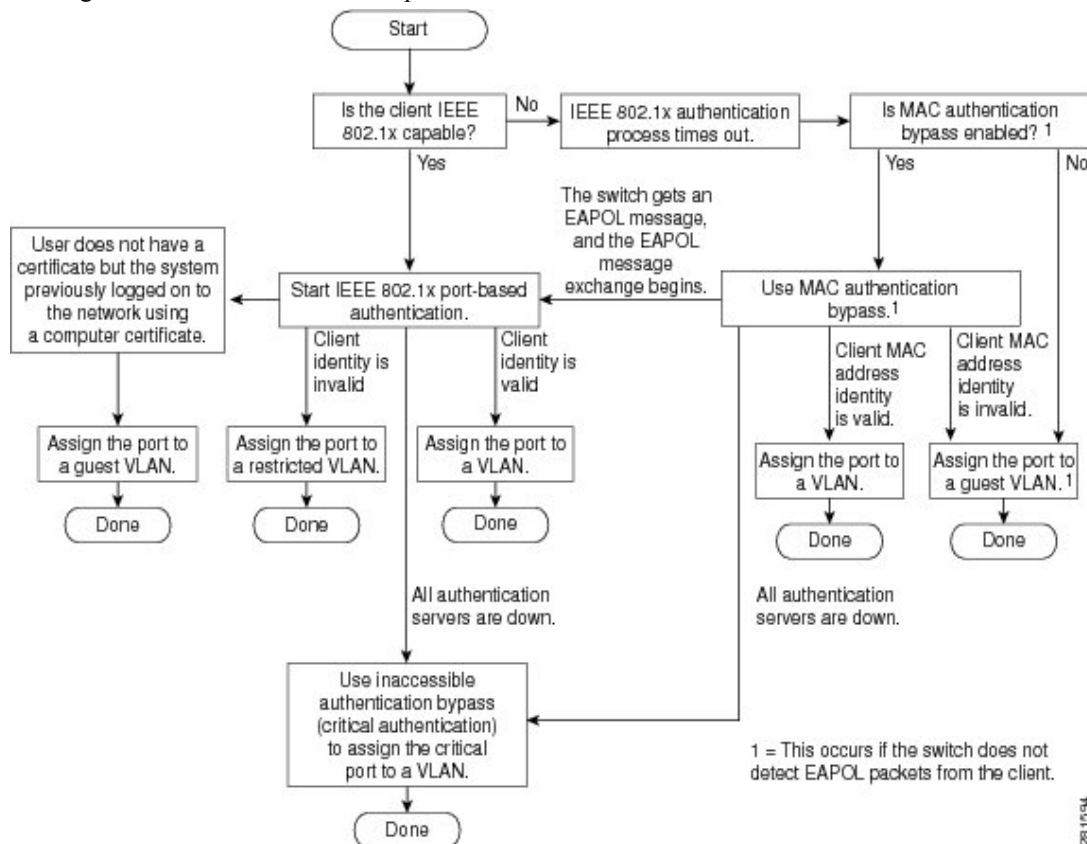


Note Inaccessible authentication bypass is also referred to as critical authentication or the AAA fail policy.

If Multi Domain Authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization.

Figure 108: Authentication Flowchart

This figure shows the authentication process.



The switch re-authenticates a client when one of these situations occurs:

- Periodic re-authentication is enabled, and the re-authentication timer expires.

You can configure the re-authentication timer to use a switch-specific value or to be based on values from the RADIUS server.

After 802.1x authentication using a RADIUS server is configured, the switch uses timers based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]).

The Session-Timeout RADIUS attribute (Attribute[27]) specifies the time after which re-authentication occurs. The range is 1 to 65535 seconds.

The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during re-authentication. The actions are *Initialize* and *ReAuthenticate*. When the *Initialize* action is set (the attribute value is *DEFAULT*), the 802.1x session ends, and connectivity is lost during re-authentication. When the *ReAuthenticate* action is set (the attribute value is *RADIUS-Request*), the session is not affected during re-authentication.

- You manually re-authenticate the client by entering the **dot1x re-authenticate interface interface-id** privileged EXEC command.

Port-Based Authentication Initiation and Message Exchange

During 802.1x authentication, the switch or the client can initiate authentication. If you enable authentication on a port by using the **authentication port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

However, if during bootup, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.



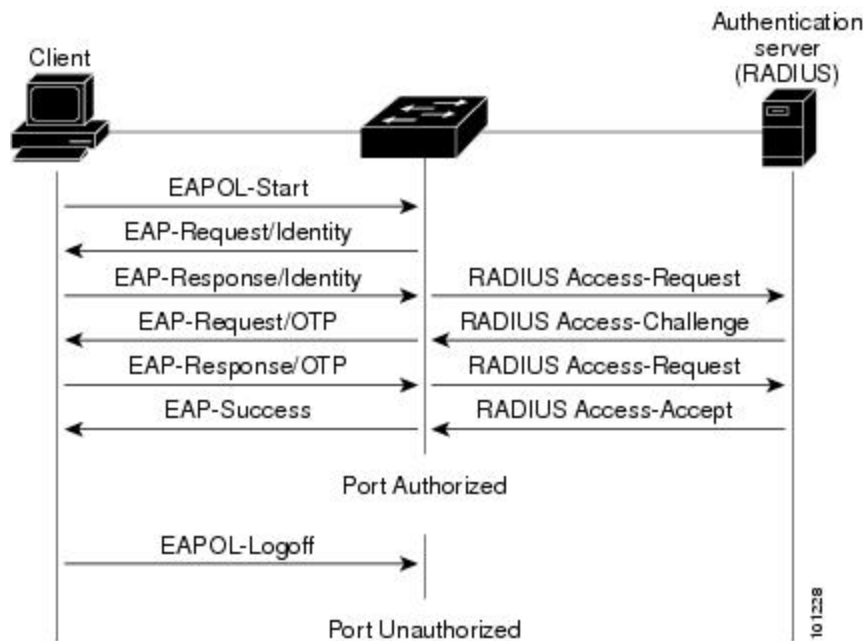
Note If 802.1x authentication is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. If the authentication fails, authentication can be retried, the port might be assigned to a VLAN that provides limited services, or network access is not granted.

The specific exchange of EAP frames depends on the authentication method being used.

Figure 109: Message Exchange

This figure shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.

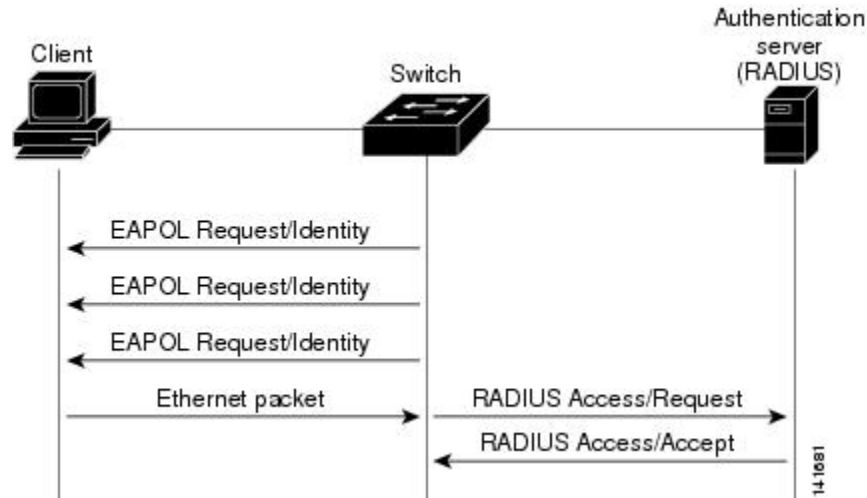


If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the

client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and starts 802.1x authentication.

Figure 110: Message Exchange During MAC Authentication Bypass

This figure shows the message exchange during MAC authentication bypass.



Authentication Manager for Port-Based Authentication

Port-Based Authentication Methods

Table 140: 802.1x Features

Authentication method	Mode			
	Single host	Multiple host	MDA	Multiple Authen
802.1x	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN Per-use Filter-I Downlo Redirec

Authentication method	Mode			
	Single host	Multiple host	MDA	Multiple Authentication
MAC authentication bypass	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL
Standalone web authentication	Proxy ACL, Filter-Id attribute, downloadable ACL			
NAC Layer 2 IP validation	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL
Web authentication as fallback method	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL

¹⁶ Supported in Cisco IOS Release 12.2(50)SE and later.

¹⁷ For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids



Note You can only set **any** as the source in the ACL.



Note For any ACL configured for multiple-host mode, the source portion of statement must be *any*. (For example, **permit icmp any host 10.10.1.1.**)



Note Using role-based ACLs as Filter-Id is not recommended.

You must specify **any** in the source ports of any defined ACL. Otherwise, the ACL cannot be applied and authorization fails. Single host is the only exception to support backward compatibility.

More than one host can be authenticated on MDA-enabled and multiauth ports. The ACL policy applied for one host does not effect the traffic of another host. If only one host is authenticated on a multi-host port, and the other hosts gain network access without authentication, the ACL policy for the first host can be applied to the other connected hosts by specifying any in the source address.

Port-Based Authentication Manager CLI Commands

The authentication-manager interface-configuration commands control all the authentication methods, such as 802.1x, MAC authentication bypass, and web authentication. The authentication manager commands determine the priority and order of authentication methods applied to a connected host.

The authentication manager commands control generic authentication features, such as host-mode, violation mode, and the authentication timer. Generic authentication commands include the **authentication host-mode**, **authentication violation**, and **authentication timer** interface configuration commands.

802.1x-specific commands begin with the **dot1x** keyword. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface.

To disable dot1x on a switch, remove the configuration globally by using the **no dot1x system-auth-control** command, and also remove it from all configured interfaces.



Note If 802.1x authentication is globally disabled, other authentication methods are still enabled on that port, such as web authentication.

The **authentication manager** commands provide the same functionality as earlier 802.1x commands.

When filtering out verbose system messages generated by the authentication manager, the filtered content typically relates to authentication success. You can also filter verbose messages for 802.1x authentication and MAB authentication. There is a separate command for each authentication method:

- The **no authentication logging verbose** global configuration command filters verbose messages from the authentication manager.
- The **no dot1x logging verbose** global configuration command filters 802.1x authentication verbose messages.
- The **no mab logging verbose** global configuration command filters MAC authentication bypass (MAB) verbose messages

Table 141: Authentication Manager Commands and Earlier 802.1x Commands

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication control-direction <i>{both in}</i>	dot1x control-direction <i>{both in}</i>	Enable 802.1x authentication with the v (WoL) feature, and configure the port c unidirectional or bidirectional.
authentication event	dot1x auth-fail vlan dot1x critical (interface configuration) dot1x guest-vlan6	Enable the restricted VLAN on a port. Enable the inaccessible-authentication-b Specify an active VLAN as an 802.1x p
authentication fallback <i>fallback-profile</i>	dot1x fallback <i>fallback-profile</i>	Configure a port to use web authentica fallback method for clients that do not s authentication.

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
<code>authentication host-mode [multi-auth multi-domain multi-host single-host]</code>	<code>dot1x host-mode {single-host multi-host multi-domain}</code>	Allow a single host (client) or multiple hosts to connect to an 802.1x-authorized port.
<code>authentication order</code>	<code>mab</code>	Provides the flexibility to define the order of authentication methods to be used.
<code>authentication periodic</code>	<code>dot1x reauthentication</code>	Enable periodic re-authentication of the client.
<code>authentication port-control {auto force-authorized force-unauthorized}</code>	<code>dot1x port-control {auto force-authorized force-unauthorized}</code>	Enable manual control of the authorization state of the port.
<code>authentication timer</code>	<code>dot1x timeout</code>	Set the 802.1x timers.
<code>authentication violation {protect restrict shutdown}</code>	<code>dot1x violation-mode {shutdown restrict protect}</code>	Configure the violation modes that occur when a client device connects to a port or when a new device connects to a port after the maximum number of devices are connected to that port.

Ports in Authorized and Unauthorized States

During 802.1x authentication, depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port that is not configured as a voice VLAN port disallows all ingress and egress traffic except for 802.1x authentication, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally. If the port is configured as a voice VLAN port, the port allows VoIP traffic and 802.1x protocol packets before the client is successfully authenticated.



Note CDP bypass is not supported and may cause a port to go into err-disabled state.

If a client that does not support 802.1x authentication connects to an unauthorized 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an 802.1x-enabled client connects to a port that is not running the 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **authentication port-control** interface configuration command and these keywords:

- **force-authorized**—disables 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without 802.1x-based authentication of the client. This is the default setting.

- **force-unauthorized**—causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.
- **auto**—enables 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a client logs off, it sends an EAPOL-logoff message, causing the switch port to change to the unauthorized state.

If the link state of a port changes from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

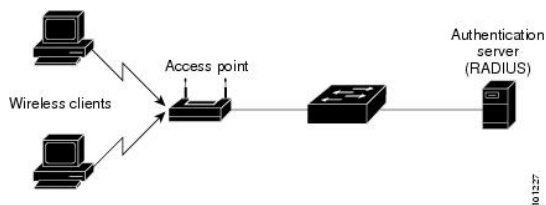
802.1x Host Mode

You can configure an 802.1x port for single-host or for multiple-hosts mode. In single-host mode, only one client can be connected to the 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single 802.1x-enabled port. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients.

In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

Figure 111: Multiple Host Mode Example



Note For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

The switch supports multidomain authentication (MDA), which allows both a data device and a voice device, such as an IP Phone (Cisco or non-Cisco), to connect to the same switch port.

802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows multiple authenticated clients on the data VLAN and voice VLAN. Each host is individually authenticated. There is no limit to the number of data or voice device that can be authenticated on a multiauthport.

If a hub or access point is connected to an 802.1x-enabled port, each connected client must be authenticated. For non-802.1x devices, you can use MAC authentication bypass or web authentication as the per-host authentication fallback method to authenticate different hosts with different methods on a single port.

You can assign a RADIUS-server-supplied VLAN in multi-auth mode, under the following conditions:

- The host is the first host authorized on the port, and the RADIUS server supplies VLAN information
- Subsequent hosts are authorized with a VLAN that matches the operational VLAN.
- A host is authorized on the port with no VLAN assignment, and subsequent hosts either have no VLAN assignment, or their VLAN information matches the operational VLAN.
- The first host authorized on the port has a group VLAN assignment, and subsequent hosts either have no VLAN assignment, or their group VLAN matches the group VLAN on the port. Subsequent hosts must use the same VLAN from the VLAN group as the first host. If a VLAN list is used, all hosts are subject to the conditions specified in the VLAN list.
- After a VLAN is assigned to a host on the port, subsequent hosts must have matching VLAN information or be denied access to the port.
- The behavior of the critical-auth VLAN is not changed for multi-auth mode. When a host tries to authenticate and the server is not reachable, all authorized hosts are reinitialized in the configured VLAN.

Multi-auth Per User VLAN assignment

The Multi-auth Per User VLAN assignment feature allows you to create multiple operational access VLANs based on VLANs assigned to the clients on the port that has a single configured access VLAN. The port configured as an access port where the traffic for all the VLANs associated with data domain is not dot1q tagged, and these VLANs are treated as native VLANs.

The number of hosts per multi-auth port is 8, however there can be more hosts.

The following scenarios are associated with the multi-auth Per User VLAN assignments:

Scenario one

When a hub is connected to an access port, and the port is configured with an access VLAN (V0).

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. This behaviour is similar on a single-host or multi-domain-auth port.

When a second host (H2) is connected and gets assigned to VLAN (V2), the port will have two operational VLANs (V1 and V2). If H1 and H2 sends untagged ingress traffic, H1 traffic is mapped to VLAN (V1) and H2 traffic to VLAN (V2), all egress traffic going out of the port on VLAN (V1) and VLAN (V2) are untagged.

If both the hosts, H1 and H2 are logged out or the sessions are removed due to some reason then VLAN (V1) and VLAN (V2) are removed from the port, and the configured VLAN (V0) is restored on the port.

Scenario two

When a hub is connected to an access port, and the port is configured with an access VLAN (V0). The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1.

When a second host (H2) is connected and gets authorized without explicit vlan policy, H2 is expected to use the configured VLAN (V0) that is restored on the port. All egress traffic going out of two operational VLANs, VLAN (V0) and VLAN (V1) are untagged.

If host (H2) is logged out or the session is removed due to some reason then the configured VLAN (V0) is removed from the port, and VLAN (V1) becomes the only operational VLAN on the port.

Scenario three

When a hub is connected to an access port in open mode, and the port is configured with an access VLAN (V0).

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. When a second host (H2) is connected and remains unauthorized, it still has access to operational VLAN (V1) due to open mode.

If host H1 is logged out or the session is removed due to some reason, VLAN (V1) is removed from the port and host (H2) gets assigned to VLAN (V0).



Note The combination of Open mode and VLAN assignment has an adverse affect on host (H2) because it has an IP address in the subnet that corresponds to VLAN (V1).

Limitation in Multi-auth Per User VLAN assignment

In the Multi-auth Per User VLAN assignment feature, egress traffic from multiple vlans are untagged on a port where the hosts receive traffic that is not meant for them. This can be a problem with broadcast and multicast traffic.

- **IPv4 ARPs:** Hosts receive ARP packets from other subnets. This is a problem if two subnets in different Virtual Routing and Forwarding (VRF) tables with overlapping IP address range are active on the port. The host ARP cache may get invalid entries.
- **IPv6 control packets:** In IPv6 deployments, Router Advertisements (RA) are processed by hosts that are not supposed to receive them. When a host from one VLAN receives RA from a different VLAN, the host assign incorrect IPv6 address to itself. Such a host is unable to get access to the network.

The workaround is to enable the IPv6 first hop security so that the broadcast ICMPv6 packets are converted to unicast and sent out from multi-auth enabled ports.. The packet is replicated for each client in multi-auth port belonging to the VLAN and the destination MAC is set to an individual client. Ports having one VLAN, ICMPv6 packets broadcast normally.

- **IP multicast:** Multicast traffic destined to a multicast group gets replicated for different VLANs if the hosts on those VLANs join the multicast group. When two hosts in different VLANs join a multicast group (on the same mutli-auth port), two copies of each multicast packet are sent out from that port.

MAC Move

When a MAC address is authenticated on one switch port, that address is not allowed on another authentication manager-enabled port of the switch. If the switch detects that same MAC address on another authentication manager-enabled port, the address is not allowed.

There are situations where a MAC address might need to move from one port to another on the same switch. For example, when there is another device (for example a hub or an IP phone) between an authenticated host

and a switch port, you might want to disconnect the host from the device and connect it directly to another port on the same switch.

You can globally enable MAC move so the device is reauthenticated on the new port. When a host moves to a second port, the session on the first port is deleted, and the host is reauthenticated on the new port. MAC move is supported on all host modes. (The authenticated host can move to any port on the switch, no matter which host mode is enabled on the that port.) When a MAC address moves from one port to another, the switch terminates the authenticated session on the original port and initiates a new authentication sequence on the new port. The MAC move feature applies to both voice and data hosts.



Note In open authentication mode, a MAC address is immediately moved from the original port to the new port, with no requirement for authorization on the new port.

MAC Replace

The MAC replace feature can be configured to address the violation that occurs when a host attempts to connect to a port where another host was previously authenticated.



Note This feature does not apply to ports in multi-auth mode, because violations are not triggered in that mode. It does not apply to ports in multiple host mode, because in that mode, only the first host requires authentication.

If you configure the **authentication violation** interface configuration command with the **replace** keyword, the authentication process on a port in multi-domain mode is:

- A new MAC address is received on a port with an existing authenticated MAC address.
- The authentication manager replaces the MAC address of the current data host on the port with the new MAC address.
- The authentication manager initiates the authentication process for the new MAC address.
- If the authentication manager determines that the new host is a voice host, the original voice host is removed.

If a port is in open authentication mode, any new MAC address is immediately added to the MAC address table.

802.1x Accounting

The 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. 802.1x accounting is disabled by default. You can enable 802.1x accounting to monitor this activity on 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.
- Link-down occurs.
- Re-authentication successfully occurs.

- Re-authentication fails.

The switch does not log 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is in the Acct-Input-Octets or the Acct-Output-Octets attributes of a RADIUS packet.)

AV pairs are automatically sent by a switch that is configured for 802.1x accounting. Three types of RADIUS accounting packets are sent by a switch:

- START—sent when a new user session starts
- INTERIM—sent during an existing session for updates
- STOP—sent when a session terminates

You can view the AV pairs that are being sent by the switch by entering the **debug radius accounting** privileged EXEC command. For more information about this command, see the *Cisco IOS Debug Command Reference, Release 12.4*.

This table lists the AV pairs and when they are sent are sent by the switch.

Table 142: Accounting AV Pairs

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[1]	User-Name	Always	Always	Always
Attribute[4]	NAS-IP-Address	Always	Always	Always
Attribute[5]	NAS-Port	Always	Always	Always
Attribute[8]	Framed-IP-Address	Never	Sometimes ¹⁸	Sometimes
Attribute[25]	Class	Always	Always	Always
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always
Attribute[47]	Acct-Input-Packets	Never	Always	Always
Attribute[48]	Acct-Output-Packets	Never	Always	Always

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

¹⁸ The Framed-IP-Address AV pair is sent when a valid static IP address is configured or when a Dynamic Host Control Protocol (DHCP) binding exists for the host in the DHCP snooping bindings table.

802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable. You use an alternate authentication such as MAC authentication bypass or web authentication for the devices that do not support 802.1x functionality.

This feature only works if the supplicant on the client supports a query with the NOTIFY EAP notification packet. The client must respond within the 802.1x timeout value.

Switch-to-RADIUS-Server Communication

RADIUS security servers are identified by their hostname or IP address, hostname and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service—for example, authentication—the second host entry configured acts as the fail-over backup to the first one. The RADIUS host entries are tried in the order that they were configured.

802.1x Authentication with VLAN Assignment

The switch supports 802.1x authentication with VLAN assignment. After successful 802.1x authentication of a port, the RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port. You can use this feature to limit network access for certain users.

Voice device authentication is supported with multidomain host mode in Cisco IOS Release 12.2(37)SE. In Cisco IOS Release 12.2(40)SE and later, when a voice device is authorized and the RADIUS server returned an authorized VLAN, the voice VLAN on the port is configured to send and receive packets on the assigned voice VLAN. Voice VLAN assignment behaves the same as data VLAN assignment on multidomain authentication (MDA)-enabled ports.

When configured on the switch and the RADIUS server, 802.1x authentication with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if 802.1x authentication is disabled, the port is configured in its access VLAN after successful authentication. Recall that an access VLAN is a VLAN assigned to an access port. All packets sent from or received on this port belong to this VLAN.
- If 802.1x authentication is enabled but the VLAN information from the RADIUS server is not valid, authorization fails and configured VLAN remains in use. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, a nonexistent or internal (routed port) VLAN ID, an RSPAN VLAN, a shut down or suspended VLAN. In the case of a multidomain host port, configuration errors can also be due to an attempted assignment of a data VLAN that matches the configured or assigned voice VLAN ID (or the reverse).

- If 802.1x authentication is enabled and all information from the RADIUS server is valid, the authorized device is placed in the specified VLAN after authentication.
- If the multiple-hosts mode is enabled on an 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- Enabling port security does not impact the RADIUS server-assigned VLAN behavior.
- If 802.1x authentication is disabled on the port, it is returned to the configured access VLAN and configured voice VLAN.
- If an 802.1x port is authenticated and put in the RADIUS server-assigned VLAN, any change to the port access VLAN configuration does not take effect. In the case of a multidomain host, the same applies to voice devices when the port is fully authorized with these exceptions:
 - If the VLAN configuration change of one device results in matching the other device configured or assigned VLAN, then authorization of all devices on the port is terminated and multidomain host mode is disabled until a valid configuration is restored where data and voice device configured VLANs no longer match.
 - If a voice device is authorized and is using a downloaded voice VLAN, the removal of the voice VLAN configuration, or modifying the configuration value to *dot1p* or *untagged* results in voice device un-authorization and the disablement of multi-domain host mode.

When the port is in the force authorized, force unauthorized, unauthorized, or shutdown state, it is put into the configured access VLAN.

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication. (The VLAN assignment feature is automatically enabled when you configure 802.1x authentication on an access port).
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
 - [64] Tunnel-Type = VLAN
 - [65] Tunnel-Medium-Type = 802
 - [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID
 - [83] Tunnel-Preference

Attribute [64] must contain the value *VLAN* (type 13). Attribute [65] must contain the value *802* (type 6). Attribute [81] specifies the *VLAN name* or *VLAN ID* assigned to the IEEE 802.1x-authenticated user.

802.1x Authentication with Per-User ACLs

You can enable per-user access control lists (ACLs) to provide different levels of network access and service to an 802.1x-authenticated user. When the RADIUS server authenticates a user connected to an 802.1x port, it retrieves the ACL attributes based on the user identity and sends them to the switch. The switch applies the attributes to the 802.1x port for the duration of the user session. The switch removes the per-user ACL configuration when the session is over, if authentication fails, or if a link-down condition occurs. The switch does not save RADIUS-specified ACLs in the running configuration. When the port is unauthorized, the switch removes the ACL from the port.

You can configure router ACLs and input port ACLs on the same switch. However, a port ACL takes precedence over a router ACL. If you apply input port ACL to an interface that belongs to a VLAN, the port ACL takes precedence over an input router ACL applied to the VLAN interface. Incoming packets received on the port, to which a port ACL is applied, are filtered by the port ACL. Incoming routed packets received on other ports are filtered by the router ACL. Outgoing routed packets are filtered by the router ACL. To avoid configuration conflicts, you should carefully plan the user profiles stored on the RADIUS server.

RADIUS supports per-user attributes, including vendor-specific attributes. These vendor-specific attributes (VSAs) are in octet-string format and are passed to the switch during the authentication process. The VSAs used for per-user ACLs are `inacl#<n>` for the ingress direction and `outacl#<n>` for the egress direction. MAC ACLs are supported only in the ingress direction. The switch supports VSAs only in the ingress direction. It does not support port ACLs in the egress direction on Layer 2 ports.

Use only the extended ACL syntax style to define the per-user configuration stored on the RADIUS server. When the definitions are passed from the RADIUS server, they are created by using the extended naming convention. However, if you use the Filter-Id attribute, it can point to a standard ACL.

You can use the Filter-Id attribute to specify an inbound or outbound ACL that is already configured on the switch. The attribute contains the ACL number followed by `.in` for ingress filtering or `.out` for egress filtering. If the RADIUS server does not allow the `.in` or `.out` syntax, the access list is applied to the outbound ACL by default. The user is marked unauthorized if the Filter-Id sent from the RADIUS server is not configured on the device. Because of limited support of Cisco IOS access lists on the switch, the Filter-Id attribute is supported only for IP ACLs numbered in the range of 1 to 199 (IP standard ACLs) and 1300 to 2699 (IP extended ACLs).

The maximum size of the per-user ACL is 4000 ASCII characters but is limited by the maximum size of RADIUS-server per-user ACLs.

You must meet the following prerequisites to configure per-user ACLs:

- Enable AAA authentication.
- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication.
- Configure the user profile and VSAs on the RADIUS server.
- Configure the 802.1x port for single-host mode.

802.1x Authentication with Downloadable ACLs and Redirect URLs



Note IPv6 does not support Redirect URLs.

You can download ACLs and redirect URLs from a RADIUS server to the switch during 802.1x authentication or MAC authentication bypass of the host. You can also download ACLs during web authentication.



Note A downloadable ACL is also referred to as a *dACL*.

If more than one host is authenticated and the host is in single-host, MDA, or multiple-authentication mode, the switch changes the source address of the ACL to the host IP address.

You can apply the ACLs and redirect URLs to all the devices connected to the 802.1x-enabled port.

If no ACLs are downloaded during 802.1x authentication, the switch applies the static default ACL on the port to the host. On a voice VLAN port configured in multi-auth or MDA mode, the switch applies the ACL only to the phone as part of the authorization policies.



Note The limit for dACL with stacking is 64 ACEs per dACL per port. The limit without stacking is the number of available TCAM entries which varies based on the other ACL features that are active.

If there is no static ACL on a port, a dynamic auth-default ACL is created, and policies are enforced before dACLs are downloaded and applied.



Note The auth-default-ACL does not appear in the running configuration.

The auth-default ACL is created when at least one host with an authorization policy is detected on the port. The auth-default ACL is removed from the port when the last authenticated session ends. You can configure the auth-default ACL for IPv4 by using the **ip access-list extended auth-default-acl** command in global configuration mode. For IPv6, use the **ipv6 access-list extended auth-default-acl** command in the global configuration mode.



Note The auth-default-ACL does not support Cisco Discovery Protocol bypass in the single host mode. You must configure a static ACL on the interface to support Cisco Discovery Protocol bypass.

The 802.1x and MAB authentication methods support two authentication modes, *open* and *closed*. If there is no static ACL on a port in *closed* authentication mode:

- An auth-default-ACL is created.
- The auth-default-ACL allows only DHCP traffic until policies are enforced.
- When the first host authenticates, the authorization policy is applied without IP address insertion.

- When a second host is detected, the policies for the first host are refreshed, and policies for the first and subsequent sessions are enforced with IP address insertion.

If there is no static ACL on a port in *open* authentication mode:

- An auth-default-ACL-OPEN is created and allows all traffic.
- Policies are enforced with IP address insertion to prevent security breaches.
- Web authentication is subject to the auth-default-ACL-OPEN.

To control access for hosts with no authorization policy, you can configure a directive. The supported values for the directive are *open* and *default*. When you configure the *open* directive, all traffic is allowed. The *default* directive subjects traffic to the access provided by the port. You can configure the directive either in the user profile on the AAA server or on the switch. To configure the directive on the AAA server, use the **authz-directive =<open/default>** global command. To configure the directive on the switch, use the **epm access-control open** global configuration command.



Note The default value of the directive is *default*.

If a host falls back to web authentication on a port without a configured ACL:

- If the port is in open authentication mode, the auth-default-ACL-OPEN is created.
- If the port is in closed authentication mode, the auth-default-ACL is created.

The access control entries (ACEs) in the fallback ACL are converted to per-user entries. If the configured fallback profile does not include a fallback ACL, the host is subject to the auth-default-ACL associated with the port.



Note If you use a custom logo with web authentication and it is stored on an external server, the port ACL must allow access to the external server before authentication. You must either configure a static port ACL or change the auth-default-ACL to provide appropriate access to the external server.

Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL

The switch uses these *cisco-av-pair* VSAs:

- url-redirect is the HTTP or HTTPS URL.
- url-redirect-acl is the switch ACL name or number.

The switch uses the CiscoSecure-defined-ACL attribute value pair to intercept an HTTP or HTTPS request from the end point. The switch then forwards the client web browser to the specified redirect address. The url-redirect AV pair on the Cisco Secure ACS contains the URL to which the web browser is redirected. The url-redirect-acl attribute value pair contains the name or number of an ACL that specifies the HTTP or HTTPS traffic to redirect.

**Note**

- Traffic that matches a permit ACE in the ACL is redirected.
- Define the URL redirect ACL and the default port ACL on the switch.

If a redirect URL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

This section describes the ACS server switchover or failover behavior:

The first authorization request is sent to the primary ACS server; after the time out period set by the `tacacs-server timeout` command ends, the request is switched-over to the secondary server for authorization. After the first authorization request, all succeeding requests are sent to the secondary ACS server. After the switchover, if the secondary server is not available, attempts are made to reach the server and after the timeout period, authorization requests are then sent to the primary ACS server. If both servers are down, authorization requests are sent to the next ACS server in the list, after the configured timeout period ends, sent to the next server, and so on. If none of the servers are reachable, the user receives an authorization failed message.

Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs

You can set the CiscoSecure-Defined-ACL Attribute-Value (AV) pair on the Cisco Secure ACS with the RADIUS `cisco-av-pair` vendor-specific attributes (VSAs). This pair specifies the names of the downloadable ACLs on the Cisco Secure ACS with the `#ACL#-IP-name-number` attribute for IPv4 and `#ACL#-.in.ipv6` attribute for IPv6.

- The *name* is the ACL name.
- The *number* is the version number (for example, 3f783768).

If a downloadable ACL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

If the default ACL is configured on the switch and the Cisco Secure ACS sends a `host-access-policy` to the switch, it applies the policy to traffic from the host connected to a switch port. If the policy does not apply, the switch applies the default ACL. If the Cisco Secure ACS sends the switch a downloadable ACL, this ACL takes precedence over the default ACL that is configured on the switch port. However, if the switch receives an host access policy from the Cisco Secure ACS but the default ACL is not configured, the authorization failure is declared.

VLAN ID-Based MAC Authentication

You can use VLAN ID-based MAC authentication if you wish to authenticate hosts based on a static VLAN ID instead of a downloadable VLAN. When you have a static VLAN policy configured on your switch, VLAN information is sent to an IAS (Microsoft) RADIUS server along with the MAC address of each host for authentication. The VLAN ID configured on the connected port is used for MAC authentication. By using VLAN ID-based MAC authentication with an IAS server, you can have a fixed number of VLANs in the network.

The feature also limits the number of VLANs monitored and handled by STP. The network can be managed as a fixed VLAN.

802.1x Authentication with Guest VLAN

You can configure a guest VLAN for each 802.1x port on the switch to provide limited services to clients, such as downloading the 802.1x client. These clients might be upgrading their system for 802.1x authentication, and some hosts, such as Windows 98 systems, might not be IEEE 802.1x-capable.

When you enable a guest VLAN on an 802.1x port, the switch assigns clients to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.

The switch maintains the EAPOL packet history. If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an IEEE 802.1x-capable supplicant, and the interface does not change to the guest VLAN state. EAPOL history is cleared if the interface link status goes down. If no EAPOL packet is detected on the interface, the interface changes to the guest VLAN state.

If the switch is trying to authorize an 802.1x-capable voice device and the AAA server is unavailable, the authorization attempt fails, but the detection of the EAPOL packet is saved in the EAPOL history. When the AAA server becomes available, the switch authorizes the voice device. However, the switch no longer allows other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the **authentication event no-response action authorize vlan** *vlan-id* interface configuration command to allow access to the guest VLAN.
- Enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command to restart the port.

If devices send EAPOL packets to the switch during the lifetime of the link, the switch no longer allows clients that fail authentication access to the guest VLAN.



Note If an EAPOL packet is detected after the interface has changed to the guest VLAN, the interface reverts to an unauthorized state, and 802.1x authentication restarts.

Any number of 802.1x-incapable clients are allowed access when the switch port is moved to the guest VLAN. If an 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on 802.1x ports in single host, multiple host, multi-auth and multi-domain modes.

You can configure any active VLAN except an RSPAN VLAN, a private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

The switch supports *MAC authentication bypass*. When MAC authentication bypass is enabled on an 802.1x port, the switch can authorize clients based on the client MAC address when IEEE 802.1x authentication times out while waiting for an EAPOL message exchange. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is specified.

802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN (also referred to as an *authentication failed VLAN*) for each IEEE 802.1x port on a switch stack or a switch to provide limited services to clients that cannot access the guest VLAN. These clients are 802.1x-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users without valid credentials in an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.



Note You can configure a VLAN to be both the guest VLAN and the restricted VLAN if you want to provide the same services to both types of users.

Without this feature, the client attempts and fails authentication indefinitely, and the switch port remains in the spanning-tree blocking state. With this feature, you can configure the switch port to be in the restricted VLAN after a specified number of authentication attempts (the default value is 3 attempts).

The authenticator counts the failed authentication attempts for the client. When this count exceeds the configured maximum number of authentication attempts, the port moves to the restricted VLAN. The failed attempt count increments when the RADIUS server replies with either an *EAP failure* or an empty response without an EAP packet. When the port moves into the restricted VLAN, the failed attempt counter resets.

Users who fail authentication remain in the restricted VLAN until the next re-authentication attempt. A port in the restricted VLAN tries to re-authenticate at configured intervals (the default is 60 seconds). If re-authentication fails, the port remains in the restricted VLAN. If re-authentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable re-authentication. If you do this, the only way to restart the authentication process is for the port to receive a *link down* or *EAP logoff* event. We recommend that you keep re-authentication enabled if a client might connect through a hub. When a client disconnects from the hub, the port might not receive the *link down* or *EAP logoff* event.

After a port moves to the restricted VLAN, a simulated EAP success message is sent to the client. This prevents clients from indefinitely attempting authentication. Some clients (for example, devices running Windows XP) cannot implement DHCP without EAP success.

Restricted VLANs are supported on 802.1x ports in all host modes and on Layer 2 ports.

You can configure any active VLAN except an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

Other security port features such as dynamic ARP Inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

802.1x Authentication with Inaccessible Authentication Bypass

Use the inaccessible authentication bypass feature, also referred to as *critical authentication* or the *AAA fail policy*, when the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated. You can configure the switch to connect those hosts to *critical ports*.

When a new host tries to connect to the critical port, that host is moved to a user-specified access VLAN, the *critical VLAN*. The administrator gives limited authentication to the hosts.

When the switch tries to authenticate a host connected to a critical port, the switch checks the status of the configured RADIUS server. If a server is available, the switch can authenticate the host. However, if all the RADIUS servers are unavailable, the switch grants network access to the host and puts the port in the *critical-authentication* state, which is a special case of the authentication state.



Note If *critical authentication* is configured on interface, then vlan used for critical authorization (*critical vlan*) should be active on the switch. If the *critical vlan* is inactive (or) down, *critical authentication* session will keep trying to enable inactive vlan and fail repeatedly. This can lead to large amount of memory holding.

Inaccessible Authentication Bypass Support on Multiple-Authentication Ports

When a port is configured on any host mode and the AAA server is unavailable, the port is then configured to multi-host mode and moved to the critical VLAN. To support this inaccessible bypass on multiple-authentication (multiauth) ports, use the **authentication event server dead action reinitialize vlan *vlan-id*** command. When a new host tries to connect to the critical port, that port is reinitialized and all the connected hosts are moved to the user-specified access VLAN.

This command is supported on all host modes.

Inaccessible Authentication Bypass Authentication Results

The behavior of the inaccessible authentication bypass feature depends on the authorization state of the port:

- If the port is unauthorized when a host connected to a critical port tries to authenticate and all servers are unavailable, the switch puts the port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
- If the port is already authorized and reauthentication occurs, the switch puts the critical port in the critical-authentication state in the current VLAN, which might be the one previously assigned by the RADIUS server.
- If the RADIUS server becomes unavailable during an authentication exchange, the current exchange times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

You can configure the critical port to reinitialize hosts and move them out of the critical VLAN when the RADIUS server is again available. When this is configured, all critical ports in the critical-authentication state are automatically re-authenticated.

Inaccessible Authentication Bypass Feature Interactions

Inaccessible authentication bypass interacts with these features:

- Guest VLAN—Inaccessible authentication bypass is compatible with guest VLAN. When a guest VLAN is enabled on 8021.x port, the features interact as follows:
 - If at least one RADIUS server is available, the switch assigns a client to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.
 - If all the RADIUS servers are not available and the client is connected to a critical port, the switch authenticates the client and puts the critical port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.

- If all the RADIUS servers are not available and the client is not connected to a critical port, the switch might not assign clients to the guest VLAN if one is configured.
- If all the RADIUS servers are not available and if a client is connected to a critical port and was previously assigned to a guest VLAN, the switch keeps the port in the guest VLAN.
- Restricted VLAN—If the port is already authorized in a restricted VLAN and the RADIUS servers are unavailable, the switch puts the critical port in the critical-authentication state in the restricted VLAN.
- 802.1x accounting—Accounting is not affected if the RADIUS servers are unavailable.
- Private VLAN—You can configure inaccessible authentication bypass on a private VLAN host port. The access VLAN must be a secondary private VLAN.
- Voice VLAN—Inaccessible authentication bypass is compatible with voice VLAN, but the RADIUS-configured or user-specified access VLAN and the voice VLAN must be different.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

802.1x Critical Voice VLAN

When an IP phone connected to a port is authenticated by the Cisco Identity Services Engine (ISE), the phone is put into the voice domain. If the ISE is not reachable, the switch cannot determine if the device is a voice device. If the server is unavailable, the phone cannot access the voice network and therefore cannot operate.

For data traffic, you can configure inaccessible authentication bypass, or critical authentication, to allow traffic to pass through on the native VLAN when the server is not available. If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network and puts the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN. When the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated, the switch connects those hosts to critical ports. A new host trying to connect to the critical port is moved to a user-specified access VLAN, the critical VLAN, and granted limited authentication.



Note Dynamic assignment of critical voice VLAN is not supported with nested service templates. It causes the device to switch between VLANs continuously in a loop.

You can enter the **authentication event server dead action authorize voice** interface configuration command to configure the critical voice VLAN feature. When the ISE does not respond, the port goes into critical authentication mode. When traffic coming from the host is tagged with the voice VLAN, the connected device (the phone) is put in the configured voice VLAN for the port. The IP phones learn the voice VLAN identification through Cisco Discovery Protocol (Cisco devices) or through LLDP or DHCP.

You can configure the voice VLAN for a port by entering the **switchport voice vlan *vlan-id*** interface configuration command.

This feature is supported in multidomain and multi-auth host modes. Although you can enter the command when the switch is in single-host or multi-host mode, the command has no effect unless the device changes to multidomain or multi-auth host mode.

802.1x User Distribution

You can configure 802.1x user distribution to load-balance users with the same group name across multiple different VLANs.

The VLANs are either supplied by the RADIUS server or configured through the switch CLI under a VLAN group name.

- Configure the RADIUS server to send more than one VLAN name for a user. The multiple VLAN names can be sent as part of the response to the user. The 802.1x user distribution tracks all the users in a particular VLAN and achieves load balancing by moving the authorized user to the least populated VLAN.
- Configure the RADIUS server to send a VLAN group name for a user. The VLAN group name can be sent as part of the response to the user. You can search for the selected VLAN group name among the VLAN group names that you configured by using the switch CLI. If the VLAN group name is found, the corresponding VLANs under this VLAN group name are searched to find the least populated VLAN. Load balancing is achieved by moving the corresponding authorized user to that VLAN.



Note The RADIUS server can send the VLAN information in any combination of VLAN-IDs, VLAN names, or VLAN groups.

802.1x User Distribution Configuration Guidelines

- Confirm that at least one VLAN is mapped to the VLAN group.
- You can map more than one VLAN to a VLAN group.
- You can modify the VLAN group by adding or deleting a VLAN.
- When you clear an existing VLAN from the VLAN group name, none of the authenticated ports in the VLAN are cleared, but the mappings are removed from the existing VLAN group.
- If you clear the last VLAN from the VLAN group name, the VLAN group is cleared.
- You can clear a VLAN group even when the active VLANs are mapped to the group. When you clear a VLAN group, none of the ports or users that are in the authenticated state in any VLAN within the group are cleared, but the VLAN mappings to the VLAN group are cleared.

IEEE 802.1x Authentication with Voice VLAN Ports

A voice VLAN port is a special access port associated with two VLAN identifiers:

- VVID to carry voice traffic to and from the IP phone. The VVID is used to configure the IP phone connected to the port.
- PVID to carry the data traffic to and from the workstation connected to the switch through the IP phone. The PVID is the native VLAN of the port.

The IP phone uses the VVID for its voice traffic, regardless of the authorization state of the port. This allows the phone to work independently of IEEE 802.1x authentication.

In single-host mode, only the IP phone is allowed on the voice VLAN. In multiple-hosts mode, additional clients can send traffic on the voice VLAN after a supplicant is authenticated on the PVID. When multiple-hosts mode is enabled, the supplicant authentication affects both the PVID and the VVID.

A voice VLAN port becomes active when there is a link, and the device MAC address appears after the first CDP message from the IP phone. Cisco IP phones do not relay CDP messages from other devices. As a result, if several IP phones are connected in series, the switch recognizes only the one directly connected to it. When IEEE 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized IP phones more than one hop away.

When IEEE 802.1x authentication is enabled on a switch port, you can configure an access port VLAN that is also a voice VLAN.

When IP phones are connected to an 802.1x-enabled switch port that is in single host mode, the switch grants the phones network access without authenticating them. We recommend that you use multidomain authentication (MDA) on the port to authenticate both a data device and a voice device, such as an IP phone



Note If you enable IEEE 802.1x authentication on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the Cisco IP phone loses connectivity to the switch for up to 30 seconds.

IEEE 802.1x Authentication with Port Security

In general, Cisco does not recommend enabling port security when IEEE 802.1x is enabled. Since IEEE 802.1x enforces a single MAC address per port (or per VLAN when MDA is configured for IP telephony), port security is redundant and in some cases may interfere with expected IEEE 802.1x operations.

IEEE 802.1x Authentication with Wake-on-LAN

The IEEE 802.1x authentication with wake-on-LAN (WoL) feature allows dormant PCs to be powered when the switch receives a specific Ethernet frame, known as the *magic packet*. You can use this feature in environments where administrators need to connect to systems that have been powered down.

When a host that uses WoL is attached through an IEEE 802.1x port and the host powers off, the IEEE 802.1x port becomes unauthorized. The port can only receive and send EAPOL packets, and WoL magic packets cannot reach the host. When the PC is powered off, it is not authorized, and the switch port is not opened.

When the switch uses IEEE 802.1x authentication with WoL, the switch forwards traffic to unauthorized IEEE 802.1x ports, including magic packets. While the port is unauthorized, the switch continues to block ingress traffic other than EAPOL packets. The host can receive packets but cannot send packets to other devices in the network.



Note If PortFast is not enabled on the port, the port is forced to the bidirectional state.

When you configure a port as unidirectional by using the **authentication control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state. The port can send packets to the host but cannot receive packets from the host.

When you configure a port as bidirectional by using the **authentication control-direction both** interface configuration command, the port is access-controlled in both directions. The port does not receive packets from or send packets to the host.

IEEE 802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address by using the MAC authentication bypass feature. For example, you can enable this feature on IEEE 802.1x ports connected to devices such as printers.

If IEEE 802.1x authentication times out while waiting for an EAPOL response from the client, the switch tries to authorize the client by using MAC authentication bypass.

When the MAC authentication bypass feature is enabled on an IEEE 802.1x port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. After detecting a client on an IEEE 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is configured. This process works for most client devices; however, it does not work for clients that use an alternate MAC address format. You can configure how MAB authentication is performed for clients with MAC addresses that deviate from the standard format or where the RADIUS configuration requires the user name and password to differ.

If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an 802.1x-capable supplicant and uses 802.1x authentication (not MAC authentication bypass) to authorize the interface. EAPOL history is cleared if the interface link status goes down.

If the switch already authorized a port by using MAC authentication bypass and detects an IEEE 802.1x supplicant, the switch does not unauthorize the client connected to the port. When re-authentication occurs, the switch uses the authentication or re-authentication methods configured on the port, if the previous session ended because the Termination-Action RADIUS attribute value is `DEFAULT`.

Clients that were authorized with MAC authentication bypass can be re-authenticated. The re-authentication process is the same as that for clients that were authenticated with IEEE 802.1x. During re-authentication, the port remains in the previously assigned VLAN. If re-authentication is successful, the switch keeps the port in the same VLAN. If re-authentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If re-authentication is based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]) and if the Termination-Action RADIUS attribute (Attribute [29]) action is *Initialize* (the attribute value is `DEFAULT`), the MAC authentication bypass session ends, and connectivity is lost during re-authentication. If MAC authentication bypass is enabled and the IEEE 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate re-authorization. For more information about these AV pairs, see RFC 3580, “IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines.”

MAC authentication bypass interacts with the features:

- IEEE 802.1x authentication—You can enable MAC authentication bypass only if 802.1x authentication is enabled on the port .
- Guest VLAN—If a client has an invalid MAC address identity, the switch assigns the client to a guest VLAN if one is configured.

- Restricted VLAN—This feature is not supported when the client connected to an IEEE 802.1x port is authenticated with MAC authentication bypass.
- Port security
- Voice VLAN
- Private VLAN—You can assign a client to a private VLAN.
- Network Edge Access Topology (NEAT)—MAB and NEAT are mutually exclusive. You cannot enable MAB when NEAT is enabled on an interface, and you should not enable NEAT when MAB is enabled on an interface.

Cisco IOS Release 12.2(55)SE and later supports filtering of verbose MAB system messages

Network Admission Control Layer 2 IEEE 802.1x Validation

The switch supports the Network Admission Control (NAC) Layer 2 IEEE 802.1x validation, which checks the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access. With NAC Layer 2 IEEE 802.1x validation, you can do these tasks:

- Download the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute[29]) from the authentication server.
- Set the number of seconds between re-authentication attempts as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and get an access policy against the client from the RADIUS server.
- Set the action to be taken when the switch tries to re-authenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is the *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the re-authentication process starts.
- Set the list of VLAN number or name or VLAN group name as the value of the Tunnel Group Private ID (Attribute[81]) and the preference for the VLAN number or name or VLAN group name as the value of the Tunnel Preference (Attribute[83]). If you do not configure the Tunnel Preference, the first Tunnel Group Private ID (Attribute[81]) attribute is picked up from the list.
- View the NAC posture token, which shows the posture of the client, by using the **show authentication** privileged EXEC command.
- Configure secondary private VLANs as guest VLANs.

Configuring NAC Layer 2 IEEE 802.1x validation is similar to configuring IEEE 802.1x port-based authentication except that you must configure a posture token on the RADIUS server.

Flexible Authentication Ordering

You can use flexible authentication ordering to configure the order of methods that a port uses to authenticate a new host. The IEEE 802.1X Flexible Authentication feature supports three authentication methods:

- dot1X—IEEE 802.1X authentication is a Layer 2 authentication method.
- mab—MAC-Authentication Bypass is a Layer 2 authentication method.
- webauth—Web authentication is a Layer 3 authentication method.

Using this feature, you can control which ports use which authentication methods, and you can control the failover sequencing of methods on those ports. For example, MAC authentication bypass and 802.1x can be the primary or secondary authentication methods, and web authentication can be the fallback method if either or both of those authentication attempts fail.

The IEEE 802.1X Flexible Authentication feature supports the following host modes:

- multi-auth—Multiauthentication allows one authentication on a voice VLAN and multiple authentications on the data VLAN.
- multi-domain—Multidomain authentication allows two authentications: one on the voice VLAN and one on the data VLAN.

Open1x Authentication

Open1x authentication allows a device access to a port before that device is authenticated. When open authentication is configured, a new host can pass traffic according to the access control list (ACL) defined on the port. After the host is authenticated, the policies configured on the RADIUS server are applied to that host.

You can configure open authentication with these scenarios:

- Single-host mode with open authentication—Only one user is allowed network access before and after authentication.
- MDA mode with open authentication—Only one user in the voice domain and one user in the data domain are allowed.
- Multiple-hosts mode with open authentication—Any host can access the network.
- Multiple-authentication mode with open authentication—Similar to MDA, except multiple hosts can be authenticated.



Note If open authentication is configured, it takes precedence over other authentication controls. This means that if you use the **authentication open** interface configuration command, the port will grant access to the host irrespective of the **authentication port-control** interface configuration command.

Multidomain Authentication

The switch supports multidomain authentication (MDA), which allows both a data device and voice device, such as an IP phone (Cisco or non-Cisco), to authenticate on the same switch port. The port is divided into a data domain and a voice domain.



Note For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

MDA does not enforce the order of device authentication. However, for best results, we recommend that a voice device is authenticated before a data device on an MDA-enabled port.

Follow these guidelines for configuring MDA:

- You must configure a switch port for MDA.
- You must configure the voice VLAN for the IP phone when the host mode is set to multidomain.
- Voice VLAN assignment on an MDA-enabled port is supported Cisco IOS Release 12.2(40)SE and later.
- To authorize a voice device, the AAA server must be configured to send a Cisco Attribute-Value (AV) pair attribute with a value of *device-traffic-class=voice*. Without this value, the switch treats the voice device as a data device.
- The guest VLAN and restricted VLAN features only apply to the data devices on an MDA-enabled port. The switch treats a voice device that fails authorization as a data device.
- If more than one device attempts authorization on either the voice or the data domain of a port, it is error disabled.
- Until a device is authorized, the port drops its traffic. Non-Cisco IP phones or voice devices are allowed into both the data and voice VLANs. The data VLAN allows the voice device to contact a DHCP server to obtain an IP address and acquire the voice VLAN information. After the voice device starts sending on the voice VLAN, its access to the data VLAN is blocked.
- A voice device MAC address that is binding on the data VLAN is not counted towards the port security MAC address limit.
- MDA can use MAC authentication bypass as a fallback mechanism to allow the switch port to connect to devices that do not support IEEE 802.1x authentication.
- When a *data* or a *voice* device is detected on a port, its MAC address is blocked until authorization succeeds. If the authorization fails, the MAC address remains blocked for 5 minutes.
- If more than five devices are detected on the *data* VLAN or more than one voice device is detected on the *voice* VLAN while a port is unauthorized, the port is error disabled.
- When a port host mode is changed from single- or multihost to multidomain mode, an authorized data device remains authorized on the port. However, a Cisco IP phone that has been allowed on the port voice VLAN is automatically removed and must be reauthenticated on that port.
- Active fallback mechanisms such as guest VLAN and restricted VLAN remain configured after a port changes from single- or multihost mode to multidomain mode.
- Switching a port host mode from multidomain to single- or multihost mode removes all authorized devices from the port.
- If a data domain is authorized first and placed in the guest VLAN, non-IEEE 802.1x-capable voice devices need to tag their packets on the voice VLAN to trigger authentication.
- We do not recommend per-user ACLs with an MDA-enabled port. An authorized device with a per-user ACL policy might impact traffic on both the voice and data VLANs of the port. If used, only one device on the port should enforce per-user ACLs.

Limiting Login for Users

The Limiting Login feature helps Network administrators to limit the login attempt of users to a network. When a user fails to successfully login to a network within a configurable number of attempts within a

configurable time limit, the user can be blocked. This feature is enabled only for local users and not for remote users. You need to configure the **aaa authentication rejected** command in global configuration mode to enable this feature.

802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)

The Network Edge Access Topology (NEAT) feature extends identity to areas outside the wiring closet (such as conference rooms). This allows any type of device to authenticate on the port.

- **802.1x switch supplicant:** You can configure a switch to act as a supplicant to another switch by using the 802.1x supplicant feature. This configuration is helpful in a scenario, where, for example, a switch is outside a wiring closet and is connected to an upstream switch through a trunk port. A switch configured with the 802.1x switch supplicant feature authenticates with the upstream switch for secure connectivity. Once the supplicant switch authenticates successfully the port mode changes from access to trunk in an authenticator switch. In a supplicant switch you must manually configure trunk when enabling CISP.
- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

In the default state, when you connect a supplicant switch to an authenticator switch that has BPDU guard enabled, the authenticator port could be error-disabled if it receives a Spanning Tree Protocol (STP) bridge protocol data unit (BPDU) packets before the supplicant switch has authenticated. Beginning with Cisco IOS Release 15.0(1)SE, you can control traffic exiting the supplicant port during the authentication period. Entering the **dot1x supplicant controlled transient** global configuration command temporarily blocks the supplicant port during authentication to ensure that the authenticator port does not shut down before authentication completes. If authentication fails, the supplicant port opens. Entering the **no dot1x supplicant controlled transient** global configuration command opens the supplicant port during the authentication period. This is the default behavior.

We strongly recommend using the **dot1x supplicant controlled transient** command on a supplicant switch when BPDU guard is enabled on the authenticator switch port with the **spanning-tree bpduguard enable** interface configuration command.



Note If you globally enable BPDU guard on the authenticator switch by using the **spanning-tree portfast bpduguard default** global configuration command, entering the **dot1x supplicant controlled transient** command does not prevent the BPDU violation.

You can enable MDA or multiauth mode on the authenticator switch interface that connects to one more supplicant switches. Multihost mode is not supported on the authenticator switch interface.

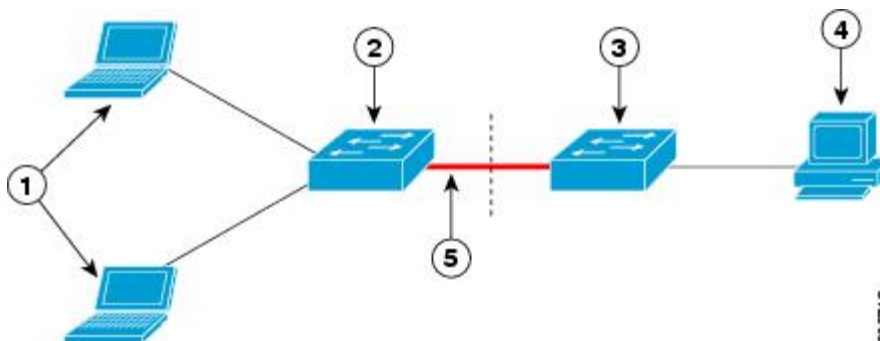
When you reboot an authenticator switch with single-host mode enabled on the interface, the interface may move to err-disabled state before authentication. To recover from err-disabled state, flap the authenticator port to activate the interface again and initiate authentication.

Use the **dot1x supplicant force-multicast** global configuration command on the supplicant switch for Network Edge Access Topology (NEAT) to work in all host modes.

- **Host Authorization:** Ensures that only traffic from authorized hosts (connecting to the switch with supplicant) is allowed on the network. The switches use Client Information Signalling Protocol (CISP) to send the MAC addresses connecting to the supplicant switch to the authenticator switch.

- Auto enablement: Automatically enables trunk configuration on the authenticator switch, allowing user traffic from multiple VLANs coming from supplicant switches. Configure the cisco-av-pair as *device-traffic-class=switch* at the ISE. (You can configure this under the *group* or the *user* settings.)

Figure 112: Authenticator and Supplicant Switch using CISP



1	Workstations (clients)	2	Supplicant switch (outside wiring closet)
3	Authenticator switch	4	Cisco ISE
5	Trunk port		



Note The **switchport nonegotiate** command is not supported on supplicant and authenticator switches with NEAT. This command should not be configured at the supplicant side of the topology. If configured on the authenticator side, the internal macros will automatically remove this command from the port.

Voice Aware 802.1x Security



Note To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

You use the voice aware 802.1x security feature to configure the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. In previous releases, when an attempt to authenticate the data client caused a security violation, the entire port shut down, resulting in a complete loss of connectivity.

You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Common Session ID

Authentication manager uses a single session ID (referred to as a common session ID) for a client no matter which authentication method is used. This ID is used for all reporting purposes, such as the show commands and MIBs. The session ID appears with all per-session syslog messages.

The session ID includes:

- The IP address of the Network Access Device (NAD)
- A monotonically increasing unique 32 bit integer
- The session start time stamp (a 32 bit integer)

This example shows how the session ID appears in the output of the show authentication command. The session ID in this example is 160000050000000B288508E5:

```
Switch# show authentication sessions
Interface  MAC Address  Method  Domain  Status  Session ID
Fa4/0/4    0000.0000.0203  mab     DATA   Authz Success  160000050000000B288508E5
```

This is an example of how the session ID appears in the syslog output. The session ID in this example is also 160000050000000B288508E5:

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

The session ID is used by the NAD, the AAA server, and other report-analyzing applications to identify the client. The ID appears automatically. No configuration is required.

How to Configure 802.1x Port-Based Authentication

Default 802.1x Authentication Configuration

Table 143: Default 802.1x Authentication Configuration

Feature	Default Setting
Switch 802.1x enable state	Disabled.
Per-port 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without 802.1x-based authentication of the client.
AAA	Disabled.

Feature	Default Setting
RADIUS server <ul style="list-style-type: none"> • IP address • UDP authentication port • Default accounting port • Key 	<ul style="list-style-type: none"> • None specified. • 1645. • 1646. • None specified.
Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic re-authentication	Disabled.
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication number	2 times (number of times that the switch restarts the authentication before the port changes to the unauthorized state).
Quiet period	60 seconds (number of seconds that the switch remains in the unauthorized state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for an EAP request/identity frame from the client before resending).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request frame before restarting the authentication process).
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before retransmitting the request to the client.)
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before retransmitting the response to the server.) You can change this timeout period by using the dot1x timeout server interface configuration command.
Inactivity timeout	Disabled.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	Disabled.
Voice-aware security	Disabled.

802.1x Authentication Configuration Guidelines

802.1x Authentication

These are the 802.1x authentication configuration guidelines:

- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- If the VLAN to which an 802.1x-enabled port is assigned changes, this change is transparent and does not affect the switch. For example, this change occurs if a port is assigned to a RADIUS server-assigned VLAN and is then assigned to a different VLAN after re-authentication.

If the VLAN to which an 802.1x port is assigned to shut down, disabled, or removed, the port becomes unauthorized. For example, the port is unauthorized after the access VLAN to which a port is assigned shuts down or is removed.

- The 802.1x protocol is supported on Layer 2 static-access ports, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:
 - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.
 - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
 - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.
- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.
- Cisco IOS Release 12.2(55)SE and later supports filtering of system messages related to 802.1x authentication.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass

These are the configuration guidelines for VLAN assignment, guest VLAN, restricted VLAN, and inaccessible authentication bypass:

- When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the 802.1x

authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the 802.1x authentication process (**authentication timer inactivity** and **authentication timer reauthentication** interface configuration commands). The amount to decrease the settings depends on the connected 802.1x client type.

- When configuring the inaccessible authentication bypass feature, follow these guidelines:
 - The feature is supported on 802.1x port in single-host mode and multihosts mode.
 - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
 - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not re-initiate the DHCP configuration process.
 - You can configure the inaccessible authentication bypass feature and the restricted VLAN on an 802.1x port. If the switch tries to re-authenticate a critical port in a restricted VLAN and all the RADIUS servers are unavailable, switch changes the port state to the critical authentication state and remains in the restricted VLAN.
 - If the CTS links are in Critical Authentication mode and the active switch reloads, the policy where SGT was configured on a device will not be available on the new active switch. This is because the internal bindings will not be synced to the standby switch in a 3750-X switch stack.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- When wireless guest clients obtains IP from foreign client VLAN instead of anchor client VLAN, you should use the **ip dhcp required** command under the WLAN configuration to force clients to issue a new DHCP request. This prevents the clients from getting an incorrect IP at anchor.
- If the wired guest clients fail to get IP address after a Cisco WLC (foreign) reload, perform a shut/no shut on the ports used by the clients to reconnect them.

MAC Authentication Bypass

These are the MAC authentication bypass configuration guidelines:

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the 802.1x authentication guidelines.
- If you disable MAC authentication bypass from a port after the port has been authorized with its MAC address, the port state is not affected.
- If the port is in the unauthorized state and the client MAC address is not the authentication-server database, the port remains in the unauthorized state. However, if the client MAC address is added to the database, the switch can use MAC authentication bypass to re-authorize the port.
- If the port is in the authorized state, the port remains in this state until re-authorization occurs.
- You can configure a timeout period for hosts that are connected by MAC authentication bypass but are inactive. The range is 1 to 65535 seconds.

Maximum Number of Allowed Devices Per Port

This is the maximum number of devices allowed on an 802.1x-enabled port:

- In single-host mode, only one device is allowed on the access VLAN. If the port is also configured with a voice VLAN, an unlimited number of Cisco IP phones can send and receive traffic through the voice VLAN.
- In multidomain authentication (MDA) mode, one device is allowed for the access VLAN, and one IP phone is allowed for the voice VLAN.
- In multihost mode, only one 802.1x supplicant is allowed on the port, but an unlimited number of non-802.1x hosts are allowed on the access VLAN. An unlimited number of devices are allowed on the voice VLAN.

Configuring 802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable.

The 802.1x readiness check is allowed on all ports that can be configured for 802.1x. The readiness check is not available on a port that is configured as **dot1x force-unauthorized**.

Follow these steps to enable the 802.1x readiness check on the switch:

Before you begin

Follow these guidelines to enable the readiness check on the switch:

- The readiness check is typically used before 802.1x is enabled on the switch.
- If you use the **dot1x test eapol-capable** privileged EXEC command without specifying an interface, all the ports on the switch stack are tested.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- The readiness check can be sent on a port that handles multiple hosts (for example, a PC that is connected to an IP phone). A syslog message is generated for each of the clients that respond to the readiness check within the timer period.

SUMMARY STEPS

1. **enable**

2. `dot1x test eapol-capable [interface interface-id]`
3. `configure terminal`
4. `dot1x test timeout timeout`
5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>dot1x test eapol-capable [interface <i>interface-id</i>]</code></p> <p>Example:</p> <pre>Switch# dot1x test eapol-capable interface gigabitethernet1/0/13 DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL capable</pre>	<p>Enables the 802.1x readiness check on the switch.</p> <p>(Optional) For <i>interface-id</i> specify the port on which to check for IEEE 802.1x readiness.</p> <p>Note If you omit the optional interface keyword, all interfaces on the switch are tested.</p>
Step 3	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 4	<p><code>dot1x test timeout <i>timeout</i></code></p> <p>Example:</p> <pre>Switch(config)# dot1x test timeout 54</pre>	<p>(Optional) Configures the timeout used to wait for EAPOL response. The range is from 1 to 65535 seconds. The default is 10 seconds.</p>
Step 5	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p><code>show running-config</code></p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Voice Aware 802.1x Security



Note To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

You use the voice aware 802.1x security feature on the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Follow these guidelines to configure voice aware 802.1x voice security on the switch:

- You enable voice aware 802.1x security by entering the **errdisable detect cause security-violation shutdown vlan** global configuration command. You disable voice aware 802.1x security by entering the **no** version of this command. This command applies to all 802.1x-configured ports in the switch.



Note If you do not include the **shutdown vlan** keywords, the entire port is shut down when it enters the error-disabled state.

- If you use the **errdisable recovery cause security-violation** global configuration command to configure error-disabled recovery, the port is automatically re-enabled. If error-disabled recovery is not configured for the port, you re-enable it by using the **shutdown** and **no shutdown** interface configuration commands.
- You can re-enable individual VLANs by using the **clear errdisable interface interface-id vlan [vlan-list]** privileged EXEC command. If you do not specify a range, all VLANs on the port are enabled.

Beginning in privileged EXEC mode, follow these steps to enable voice aware 802.1x security:

SUMMARY STEPS

1. **configure terminal**
2. **errdisable detect cause security-violation shutdown vlan**
3. **errdisable recovery cause security-violation**
4. **clear errdisable interface interface-id vlan [vlan-list]**
5. Enter the following:
 - **shutdown**
 - **no shutdown**
6. **end**

7. show errdisable detect

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>errdisable detect cause security-violation shutdown vlan</code>	Shut down any VLAN on which a security violation error occurs. Note If the shutdown vlan keywords are not included, the entire port enters the error-disabled state and shuts down.
Step 3	<code>errdisable recovery cause security-violation</code>	Enter global configuration mode.
Step 4	<code>clear errdisable interface interface-id vlan [vlan-list]</code>	(Optional) Reenable individual VLANs that have been error disabled. <ul style="list-style-type: none"> • For interface-id specify the port on which to reenable individual VLANs. • (Optional) For vlan-list specify a list of VLANs to be re-enabled. If vlan-list is not specified, all VLANs are re-enabled.
Step 5	Enter the following: <ul style="list-style-type: none"> • <code>shutdown</code> • <code>no shutdown</code> 	(Optional) Re-enable an error-disabled VLAN, and clear all error-disable indications.
Step 6	<code>end</code>	Return to privileged EXEC mode.
Step 7	<code>show errdisable detect</code>	Verify your entries.

Example

This example shows how to configure the switch to shut down any VLAN on which a security violation error occurs:

```
Switch(config)# errdisable detect cause security-violation shutdown vlan
```

This example shows how to re-enable all VLANs that were error disabled on port Gigabit Ethernet 40/2.

```
Switch# clear errdisable interface gigabitethernet40/2
vlan
```

You can verify your settings by entering the `show errdisable detect` privileged EXEC command.

Configuring 802.1x Violation Modes

You can configure an 802.1x port so that it shuts down, generates a syslog error, or discards packets from a new device when:

- a device connects to an 802.1x-enabled port
- the maximum number of allowed about devices have been authenticated on the port

Beginning in privileged EXEC mode, follow these steps to configure the security violation actions on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x {default} *method1***
4. **interface *interface-id***
5. **switchport mode access**
6. **authentication violation {shutdown | restrict | protect | replace}**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	aaa new-model Example: Switch(config)# aaa new-model	Enables AAA.
Step 3	aaa authentication dot1x {default} <i>method1</i> Example: Switch(config)# aaa authentication dot1x default group radius	Creates an 802.1x authentication method list. To create a default list that is used when a named list is <i>not</i> specified in the authentication command, use the default keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports. For <i>method1</i> , enter the group radius keywords to use the list of all RADIUS servers for authentication.
Step 4	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/4	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 5	switchport mode access Example:	Sets the port to access mode.

	Command or Action	Purpose
	<code>Switch(config-if)# switchport mode access</code>	
Step 6	<p>authentication violation {shutdown restrict protect replace}</p> <p>Example:</p> <pre>Switch(config-if)# authentication violation restrict</pre>	<p>Configures the violation mode. The keywords have these meanings:</p> <ul style="list-style-type: none"> • shutdown—Error disable the port. • restrict—Generate a syslog error. • protect—Drop packets from any new device that sends traffic to the port. • replace—Removes the current session and authenticates with the new host.
Step 7	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring 802.1x Authentication

To allow per-user ACLs or VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the 802.1x AAA process:

Before you begin

To configure 802.1x port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

SUMMARY STEPS

1. A user connects to a port on the switch.
2. Authentication is performed.
3. VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.
4. The switch sends a start message to an accounting server.
5. Re-authentication is performed, as necessary.
6. The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.
7. The user disconnects from the port.
8. The switch sends a stop message to the accounting server.

DETAILED STEPS

	Command or Action	Purpose
Step 1	A user connects to a port on the switch.	
Step 2	Authentication is performed.	
Step 3	VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.	
Step 4	The switch sends a start message to an accounting server.	
Step 5	Re-authentication is performed, as necessary.	
Step 6	The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.	
Step 7	The user disconnects from the port.	
Step 8	The switch sends a stop message to the accounting server.	

Configuring 802.1x Port-Based Authentication

Beginning in privileged EXEC mode, follow these steps to configure 802.1x port-based authentication:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x {default} *method1***
4. **dot1x system-auth-control**
5. **aaa authorization network {default} group radius**
6. **radius-server host *ip-address***
7. **radius-server key *string***
8. **interface *interface-id***
9. **switchport mode access**
10. **authentication port-control auto**
11. **dot1x pae authenticator**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	aaa new-model Example: <pre>Switch(config)# aaa new-model</pre>	Enables AAA.
Step 3	aaa authentication dot1x {default} method1 Example: <pre>Switch(config)# aaa authentication dot1x default group radius</pre>	<p>Creates an 802.1x authentication method list.</p> <p>To create a default list that is used when a named list is <i>not</i> specified in the authentication command, use the default keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports.</p> <p>For <i>method1</i>, enter the group radius keywords to use the list of all RADIUS servers for authentication.</p> <p>Note Though other keywords are visible in the command-line help string, only the group radius keywords are supported.</p>
Step 4	dot1x system-auth-control Example: <pre>Switch(config)# dot1x system-auth-control</pre>	Enables 802.1x authentication globally on the switch.
Step 5	aaa authorization network {default} group radius Example: <pre>Switch(config)# aaa authorization network default group radius</pre>	(Optional) Configures the switch to use user-RADIUS authorization for all network-related service requests, such as per-user ACLs or VLAN assignment.
Step 6	radius-server host ip-address Example: <pre>Switch(config)# radius-server host 124.2.2.12</pre>	(Optional) Specifies the IP address of the RADIUS server.
Step 7	radius-server key string Example: <pre>Switch(config)# radius-server key abc1234</pre>	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.

	Command or Action	Purpose
Step 8	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/2</pre>	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 9	switchport mode access Example: <pre>Switch(config-if)# switchport mode access</pre>	(Optional) Sets the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.
Step 10	authentication port-control auto Example: <pre>Switch(config-if)# authentication port-control auto</pre>	Enables 802.1x authentication on the port.
Step 11	dot1x pae authenticator Example: <pre>Switch(config-if)# dot1x pae authenticator</pre>	Sets the interface Port Access Entity to act only as an authenticator and ignore messages meant for a supplicant.
Step 12	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring the Switch-to-RADIUS-Server Communication

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, the **radius-server retransmit**, and the **radius-server key** global configuration commands.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

Before you begin

You must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

SUMMARY STEPS

1. enable
2. configure terminal
3. radius-server host {hostname | ip-address} auth-port port-number key string
4. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	radius-server host {hostname ip-address} auth-port port-number key string Example: <pre>Switch(config)# radius-server host 125.5.5.43 auth-port 1645 key rad123</pre>	Configures the RADIUS server parameters. For <i>hostname ip-address</i> , specify the server name or IP address of the remote RADIUS server. For auth-port port-number , specify the UDP destination port for authentication requests. The default is 1645. The range is 0 to 65536. For key string , specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. Note Always configure the key as the last item in the radius-server host command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. If you want to use multiple RADIUS servers, re-enter this command.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an IEEE 802.1x-authorized port that has the **authentication port-control** interface configuration command set to **auto**. Use the **multi-domain** keyword to configure and enable multidomain authentication (MDA), which allows both a host and a voice device, such as an IP phone (Cisco or non-Cisco), on the same switch port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **authentication host-mode [multi-auth | multi-domain | multi-host | single-host]**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	Specifies the port to which multiple hosts are indirectly attached, and enter interface configuration mode.
Step 3	authentication host-mode [multi-auth multi-domain multi-host single-host] Example: <pre>Switch(config-if)# authentication host-mode multi-host</pre>	<p>Allows multiple hosts (clients) on an 802.1x-authorized port.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • multi-auth—Allow multiple authenticated clients on both the voice VLAN and data VLAN. <p>Note The multi-auth keyword is only available with the authentication host-mode command.</p> <ul style="list-style-type: none"> • multi-host—Allow multiple hosts on an 802.1x-authorized port after a single host has been authenticated. • multi-domain—Allow both a host and a voice device, such as an IP phone (Cisco or non-Cisco), to be authenticated on an IEEE 802.1x-authorized port.

	Command or Action	Purpose
		<p>Note You must configure the voice VLAN for the IP phone when the host mode is set to multi-domain.</p> <p>Make sure that the authentication port-control interface configuration command is set to auto for the specified interface.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring Periodic Re-Authentication

You can enable periodic 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between attempts is 3600.

Beginning in privileged EXEC mode, follow these steps to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication periodic**
4. **authentication timer** {{{inactivity | reauthenticate | restart | unauthorized}} {value}}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	<p>authentication periodic</p> <p>Example:</p>	Enables periodic re-authentication of the client, which is disabled by default.

	Command or Action	Purpose
	Switch(config-if)# authentication periodic	Note The default value is 3600 seconds. To change the value of the reauthentication timer or to have the switch use a RADIUS-provided session timeout, enter the authentication timer reauthenticate command.
Step 4	authentication timer {[inactivity reauthenticate restart unauthorized]} {value} Example: Switch(config-if)# authentication timer reauthenticate 180	Sets the number of seconds between re-authentication attempts. The authentication timer keywords have these meanings: <ul style="list-style-type: none"> • inactivity—Interval in seconds after which if there is no activity from the client then it is unauthorized • reauthenticate—Time in seconds after which an automatic re-authentication attempt is initiated • restart value—Interval in seconds after which an attempt is made to authenticate an unauthorized port • unauthorized value—Interval in seconds after which an unauthorized session will get deleted This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time and then tries again. The **authentication timer restart** interface configuration command controls the idle period. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a number smaller than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer restart** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication timer restart <i>seconds</i> Example: Switch(config-if)# <code>authentication timer restart 30</code>	Sets the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example: Switch# <code>show authentication sessions interface gigabitethernet2/0/1</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer reauthenticate** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication timer reauthenticate <i>seconds</i> Example: Switch(config-if)# authentication timer reauthenticate 60	Sets the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 5.
Step 4	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example: Switch# show authentication sessions interface	Verifies your entries.

	Command or Action	Purpose
	<code>gigabitethernet2/0/1</code>	
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP-request/identity frame (assuming no response is received) to the client before restarting the authentication process.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x max-reauth-req** *count*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet2/0/1</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	dot1x max-reauth-req <i>count</i> Example:	Sets the number of times that the switch sends an EAP-request/identity frame to the client before restarting

	Command or Action	Purpose
	Switch(config-if)# <code>dot1x max-reauth-req 5</code>	the authentication process. The range is 1 to 10; the default is 2.
Step 4	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the re-authentication number. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **dot1x max-req** *count*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example:	Sets the port to access mode only if you previously configured the RADIUS server.

	Command or Action	Purpose
	<code>Switch(config-if)# switchport mode access</code>	
Step 4	dot1x max-req count Example: <code>Switch(config-if)# dot1x max-req 4</code>	Sets the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 0 to 10; the default is 2.
Step 5	end Example: <code>Switch(config-if)# end</code>	Returns to privileged EXEC mode.

Enabling MAC Move

MAC move allows an authenticated host to move from one port on the switch to another.

Beginning in privileged EXEC mode, follow these steps to globally enable MAC move on the switch. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **authentication mac-move permit**
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 2	authentication mac-move permit Example: <code>Switch(config)# authentication mac-move permit</code>	<p>Enables MAC move on the switch. Default is deny.</p> <p>In Session Aware Networking mode, the default CLI is access-session mac-move deny. To enable Mac Move in Session Aware Networking, use the no access-session mac-move global configuration command.</p> <p>In legacy mode (IBNS 1.0), default value for mac-move is deny and in C3PL mode (IBNS 2.0) default value is permit.</p>

	Command or Action	Purpose
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 4	show running-config Example: Switch# show running-config	Verifies your entries.
Step 5	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Disabling MAC Move

To disable MAC move from a secure port to an unsecured port on a switch, beginning in privileged EXEC mode, follow these steps. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **authentication mac-move deny-uncontrolled**
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	authentication mac-move deny-uncontrolled Example: Switch(config)# authentication mac-move deny-uncontrolled	Disables MAC move on the switch.

	Command or Action	Purpose
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 5	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling MAC Replace

MAC replace allows a host to replace an authenticated host on a port.

Beginning in privileged EXEC mode, follow these steps to enable MAC replace on an interface. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication violation** {**protect** | **replace** | **restrict** | **shutdown**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example:	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Switch(config)# <code>interface gigabitethernet2/0/2</code>	
Step 3	authentication violation {protect replace restrict shutdown} Example: Switch(config-if)# <code>authentication violation replace</code>	Use the replace keyword to enable MAC replace on the interface. The port removes the current session and initiates authentication with the new host. The other keywords have these effects: <ul style="list-style-type: none"> • protect: the port drops packets with unexpected MAC addresses without generating a system message. • restrict: violating packets are dropped by the CPU and a system message is generated. • shutdown: the port is error disabled when it receives an unexpected MAC address.
Step 4	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Accounting

Enabling AAA system accounting with 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active 802.1x sessions are closed.



Note In Cisco IOS XE Denali 16.3.x and Cisco IOS XE Everest 16.6.x, periodic AAA accounting updates are not supported. The switch does not send periodic interim accounting records to the accounting server. Periodic AAA accounting updates are available in Cisco IOS XE Fuji 16.9.x and later releases.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

Accounting message %s for session %s failed to receive Accounting Response.

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```



Note You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps. To turn on these functions, enable logging of “Update/Watchdog packets from this AAA client” in your RADIUS server Network Configuration tab. Next, enable “CVS RADIUS Accounting” in your RADIUS server System Configuration tab.

Beginning in privileged EXEC mode, follow these steps to configure 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **aaa accounting dot1x default start-stop group radius**
4. **aaa accounting system default start-stop group radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/3</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	aaa accounting dot1x default start-stop group radius Example: <pre>Switch(config-if)# aaa accounting dot1x default start-stop group radius</pre>	Enables 802.1x accounting using the list of all RADIUS servers.

	Command or Action	Purpose
Step 4	aaa accounting system default start-stop group radius Example: <pre>Switch(config-if)# aaa accounting system default start-stop group radius</pre>	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Guest VLAN

When you configure a guest VLAN, clients that are not 802.1x-capable are put into the guest VLAN when the server does not receive a response to its EAP request/identity frame. Clients that are 802.1x-capable but that fail authentication are not granted network access. The switch supports guest VLANs in single-host or multiple-hosts mode.

Beginning in privileged EXEC mode, follow these steps to configure a guest VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. Use one of the following:
 - **switchport mode access**
 - **switchport mode private-vlan host**
4. **authentication event no-response action authorize vlan *vlan-id***
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 2/0/2</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: <pre>Switch(config-if)# switchport mode private-vlan host</pre>	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	authentication event no-response action authorize vlan <i>vlan-id</i> Example: <pre>Switch(config-if)# authentication event no-response action authorize vlan 2</pre>	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch stack or a switch, clients that are IEEE 802.1x-compliant are moved into the restricted VLAN when the authentication server does not receive a valid username and password. The switch supports restricted VLANs only in single-host mode.

Beginning in privileged EXEC mode, follow these steps to configure a restricted VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**

2. **interface** *interface-id*
3. Use one of the following:
 - **switchport mode access**
 - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 2/0/2	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: Switch(config-if)# switchport mode access	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	authentication port-control auto Example: Switch(config-if)# authentication port-control auto	Enables 802.1x authentication on the port.
Step 5	authentication event fail action authorize vlan <i>vlan-id</i> Example: Switch(config-if)# authentication event fail action authorize vlan 2	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	<code>Switch(config-if)# end</code>	

Configuring Number of Authentication Attempts on a Restricted VLAN

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **authentication event retry** *retry count* interface configuration command. The range of allowable authentication attempts is 1 to 3. The default is 3 attempts.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of allowed authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **switchport mode access**
 - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **authentication event retry** *retry count*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <code>Switch(config)# interface gigabitethernet 2/0/3</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: or	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.

	Command or Action	Purpose
	Switch(config-if)# switchport mode access	
Step 4	authentication port-control auto Example: Switch(config-if)# authentication port-control auto	Enables 802.1x authentication on the port.
Step 5	authentication event fail action authorize vlan <i>vlan-id</i> Example: Switch(config-if)# authentication event fail action authorize vlan 8	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	authentication event retry <i>retry count</i> Example: Switch(config-if)# authentication event retry 2	Specifies a number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 3, and the default is 3.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN

Beginning in privileged EXEC mode, follow these steps to configure critical voice VLAN on a port and enable the inaccessible authentication bypass feature.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **radius-server dead-criteria {time *seconds* } [*tries number*]**
4. **radius-server deadtime *minutes***
5. **radius-server host ip-address *address* [*acct-port udp-port*] [*auth-port udp-port*] [*testusername name*] [*idle-time time*] [*ignore-acct-port*] [*ignore auth-port*] [*key string*]**
6. **dot1x critical {eapol | recovery delay *milliseconds*}**
7. **interface *interface-id***
8. **authentication event server dead action {authorize | reinitialize} vlan *vlan-id***
9. **switchport voice vlan *vlan-id***

10. authentication event server dead action authorize voice
11. show authentication interface *interface-id*
12. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>aaa new-model</p> <p>Example:</p> <pre>Switch(config)# aaa new-model</pre>	Enables AAA.
Step 3	<p>radius-server dead-criteria {time <i>seconds</i> } [tries <i>number</i>]</p> <p>Example:</p> <pre>Switch(config)# radius-server dead-criteria time 20 tries 10</pre>	<p>Sets the conditions that determine when a RADIUS server is considered un-available or down (dead).</p> <ul style="list-style-type: none"> • time— 1 to 120 seconds. The switch dynamically determines a default <i>seconds</i> value between 10 and 60. • number—1 to 100 tries. The switch dynamically determines a default <i>triesnumber</i> between 10 and 100.
Step 4	<p>radius-server deadtime <i>minutes</i></p> <p>Example:</p> <pre>Switch(config)# radius-server deadtime 60</pre>	(Optional) Sets the number of minutes during which a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 5	<p>radius-server host ip-address <i>address</i> [acct-port <i>udp-port</i>] [auth-port <i>udp-port</i>] [testusername <i>name</i> [idle-time <i>time</i>] [ignore-acct-port] [ignore auth-port]] [key <i>string</i>]</p> <p>Example:</p> <pre>Switch(config)# radius-server host 10.0.0.10 acct-port 1550 auth-port 1560 test username user1 idle-time 30 key abc1234</pre>	<p>(Optional) Configure the RADIUS server parameters by using these keywords:</p> <ul style="list-style-type: none"> • acct-port <i>udp-port</i>—Specify the UDP port for the RADIUS accounting server. The range for the UDP port number is from 0 to 65536. The default is 1646. • auth-port <i>udp-port</i>—Specify the UDP port for the RADIUS authentication server. The range for the UDP port number is from 0 to 65536. The default is 1645.

	Command or Action	Purpose
		<p>Note You should configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values.</p> <ul style="list-style-type: none"> • test username <i>name</i>—Enable automated testing of the RADIUS server status, and specify the username to be used. • idle-time <i>time</i>—Set the interval of time in minutes after which the switch sends test packets to the server. The range is from 1 to 35791 minutes. The default is 60 minutes (1 hour). • ignore-acct-port—Disable testing on the RADIUS-server accounting port. • ignore-auth-port—Disable testing on the RADIUS-server authentication port. • For keystring, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. <p>Note Always configure the key as the last item in the radius-server host command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.</p> <p>You can also configure the authentication and encryption key by using the radius-server key {0string 7string string} global configuration command.</p>
Step 6	<p>dot1x critical {eapol recovery delay <i>milliseconds</i>}</p> <p>Example:</p> <pre>Switch(config)# dot1x critical eapol (config)# dot1x critical recovery delay 2000</pre>	<p>(Optional) Configure the parameters for inaccessible authentication bypass:</p> <ul style="list-style-type: none"> • eapol—Specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. • recovery delay <i>milliseconds</i>—Set the recovery delay period during which the switch waits to re-initialize a critical port when a RADIUS server that was

	Command or Action	Purpose
		unavailable becomes available. The range is from 1 to 10000 milliseconds. The default is 1000 milliseconds (a port can be re-initialized every second).
Step 7	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specify the port to be configured, and enter interface configuration mode.
Step 8	authentication event server dead action { authorize reinitialize } vlan <i>vlan-id</i> Example: Switch(config-if)# authentication event server dead action reinitialicze vlan 20	Use these keywords to move hosts on the port if the RADIUS server is unreachable: <ul style="list-style-type: none"> • authorize—Move any new hosts trying to authenticate to the user-specified critical VLAN. • reinitialize—Move all authorized hosts on the port to the user-specified critical VLAN.
Step 9	switchport voice vlan <i>vlan-id</i> Example: Switch(config-if)# switchport voice vlan	Specifies the voice VLAN for the port. The voice VLAN cannot be the same as the critical data VLAN configured in Step 6.
Step 10	authentication event server dead action authorize voice Example: Switch(config-if)# authentication event server dead action authorize voice	Configures critical voice VLAN to move data traffic on the port to the voice VLAN if the RADIUS server is unreachable.
Step 11	show authentication interface <i>interface-id</i> Example: Switch(config-if)# do show authentication interface gigabit 1/0/1	(Optional) Verify your entries.
Step 12	copy running-config startup-config Example: Switch(config-if)# do copy running-config startup-config	(Optional) Verify your entries.

Example

To return to the RADIUS server default settings, use the **no radius-server dead-criteria**, the **no radius-server deadtime**, and the **no radius-server host** global configuration commands. To disable inaccessible authentication bypass, use the **no authentication event server dead action** interface configuration command. To disable critical voice VLAN, use the **no authentication event server dead action authorize voice** interface configuration command.

Example of Configuring Inaccessible Authentication Bypass

This example shows how to configure the inaccessible authentication bypass feature:

```
Switch(config)# radius-server dead-criteria time 30 tries 20
Switch(config)# radius-server deadtime 60
Switch(config)# radius-server host 10.0.0.10 acct-port 1550 auth-port 1560 test username
user1 idle-time 30 key abc1234
Switch(config)# dot1x critical eapol
Switch(config)# dot1x critical recovery delay 2000
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# dot1x critical
Switch(config-if)# dot1x critical recovery action reinitialize
Switch(config-if)# dot1x critical vlan 20
Switch(config-if)# end
```

Configuring 802.1x Authentication with WoL

Beginning in privileged EXEC mode, follow these steps to enable 802.1x authentication with WoL. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication control-direction** {both | in}
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example:	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	<code>Switch(config)# interface gigabitethernet2/0/3</code>	
Step 3	authentication control-direction {both in} Example: <code>Switch(config-if)# authentication control-direction both</code>	Enables 802.1x authentication with WoL on the port, and use these keywords to configure the port as bidirectional or unidirectional. <ul style="list-style-type: none"> • both—Sets the port as bidirectional. The port cannot receive packets from or send packets to the host. By default, the port is bidirectional. • in—Sets the port as unidirectional. The port can send packets to the host but cannot receive packets from the host.
Step 4	end Example: <code>Switch(config-if)# end</code>	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface interface-id Example: <code>Switch# show authentication sessions interface gigabitethernet2/0/3</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring MAC Authentication Bypass

Beginning in privileged EXEC mode, follow these steps to enable MAC authentication bypass. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface interface-id**
3. **authentication port-control auto**
4. **mab [eap]**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication port-control auto Example: Switch(config-if)# authentication port-control auto	Enables 802.1x authentication on the port.
Step 4	mab [eap] Example: Switch(config-if)# mab	Enables MAC authentication bypass. (Optional) Use the eap keyword to configure the switch to use EAP for authorization.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Formatting a MAC Authentication Bypass Username and Password

Use the optional **mab request format** command to format the MAB username and password in a style accepted by the authentication server. The username and password are usually the MAC address of the client. Some authentication server configurations require the password to be different from the username.

Beginning in privileged EXEC mode, follow these steps to format MAC authentication bypass username and passwords.

SUMMARY STEPS

1. **configure terminal**
2. **mab request format attribute 1 groupsize {1 | 2 | 4 | 12} [separator {- | : | .} {lowercase | uppercase}]**
3. **mab request format attribute2 {0 | 7} *text***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	mab request format attribute 1 groupsize {1 2 4 12} [separator {- : .} {lowercase uppercase}] Example: <pre>Switch(config)# mab request format attribute 1 groupsize 12</pre>	<p>Specifies the format of the MAC address in the User-Name attribute of MAB-generated Access-Request packets.</p> <p>1—Sets the username format of the 12 hex digits of the MAC address.</p> <p>group size—The number of hex nibbles to concatenate before insertion of a separator. A valid groupsize must be either 1, 2, 4, or 12.</p> <p>separator—The character that separates the hex nibbles according to group size. A valid separator must be either a hyphen, colon, or period. No separator is used for a group size of 12.</p> <p>{lowercase uppercase}—Specifies if nonnumeric hex nibbles should be in lowercase or uppercase.</p>
Step 3	mab request format attribute2 {0 7} text Example: <pre>Switch(config)# mab request format attribute 2 7 A02f44E18B12</pre>	<p>2—Specifies a custom (nondefault) value for the User-Password attribute in MAB-generated Access-Request packets.</p> <p>0—Specifies a cleartext password to follow.</p> <p>7—Specifies an encrypted password to follow.</p> <p>text—Specifies the password to be used in the User-Password attribute.</p> <p>Note When you send configuration information in e-mail, remove type 7 password information. The show tech-support command removes this information from its output by default.</p>
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring 802.1x User Distribution

Beginning in privileged EXEC mode, follow these steps to configure a VLAN group and to map a VLAN to it:

SUMMARY STEPS

1. **configure terminal**
2. **vlan group** *vlan-group-name* **vlan-list** *vlan-list*
3. **end**
4. **no vlan group** *vlan-group-name* **vlan-list** *vlan-list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> Example: Switch(config)# vlan group <i>eng-dept</i> vlan-list 10	Configures a VLAN group, and maps a single VLAN or a range of VLANs to it.
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 4	no vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> Example: Switch(config)# no vlan group <i>eng-dept</i> vlan-list 10	Clears the VLAN group configuration or elements of the VLAN group configuration.

Example of Configuring VLAN Groups

This example shows how to configure the VLAN groups, to map the VLANs to the groups, to and verify the VLAN group configurations and mapping to the specified VLANs:

```
Switch(config)# vlan group eng-dept vlan-list 10

Switch(config)# show vlan group group-name eng-dept
Group Name                Vlans Mapped
-----                -
eng-dept                    10

Switch(config)# show dot1x vlan-group all
Group Name                Vlans Mapped
-----                -
eng-dept                    10
```



```
hr-dept                               20
```

This example shows how to add a VLAN to an existing VLAN group and to verify that the VLAN was added:

```
Switch(config)# vlan group eng-dept vlan-list 30
Switch(config)# show vlan group eng-dept
Group Name                               Vlans Mapped
-----
eng-dept                                 10,30
```

This example shows how to remove a VLAN from a VLAN group:

```
Switch# no vlan group eng-dept vlan-list 10
```

This example shows that when all the VLANs are cleared from a VLAN group, the VLAN group is cleared:

```
Switch(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.
Switch(config)# show vlan group group-name eng-dept
```

This example shows how to clear all the VLAN groups:

```
Switch(config)# no vlan group eng-dept vlan-list all
Switch(config)# show vlan-group all
```

For more information about these commands, see the *Cisco IOS Security Command Reference*.

Configuring NAC Layer 2 802.1x Validation

You can configure NAC Layer 2 802.1x validation, which is also referred to as 802.1x authentication with a RADIUS server.

Beginning in privileged EXEC mode, follow these steps to configure NAC Layer 2 802.1x validation. The procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **authentication periodic**
6. **authentication timer reauthenticate**
7. **end**
8. **show authentication sessions interface** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet2/0/3	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Switch(config-if)# switchport mode access	Sets the port to access mode only if you configured the RADIUS server.
Step 4	authentication event no-response action authorize vlan <i>vlan-id</i> Example: Switch(config-if)# authentication event no-response action authorize vlan 8	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, or a voice VLAN as an 802.1x guest VLAN.
Step 5	authentication periodic Example: Switch(config-if)# authentication periodic	Enables periodic re-authentication of the client, which is disabled by default.
Step 6	authentication timer reauthenticate Example: Switch(config-if)# authentication timer reauthenticate	Sets re-authentication attempt for the client (set to one hour). This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 8	show authentication sessions interface <i>interface-id</i> Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show authentication sessions interface gigabitethernet2/0/3</code>	
Step 9	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Limiting Login for Users

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `aaa new-model`
4. `aaa authentication login default local`
5. `aaa authentication rejected n in m ban x`
6. `end`
7. `show aaa local user blocked`
8. `clear aaa local user blocked username username`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# <code>aaa new-model</code>	Enables the authentication, authorization, and accounting (AAA) access control model.
Step 4	aaa authentication login default local Example: Device(config)# <code>aaa authentication login default local</code>	Sets the authentication, authorization, and accounting (AAA) authentication by using the default authentication methods.

	Command or Action	Purpose
Step 5	aaa authentication rejected <i>n</i> in <i>m</i> ban <i>x</i> Example: <pre>Device(config)# aaa authentication rejected 3 in 20 ban 300</pre>	Configures the time period for which an user is blocked, if the user fails to successfully login within the specified time and login attempts. <ul style="list-style-type: none"> • <i>n</i>—Specifies the number of times a user can try to login. • <i>m</i>—Specifies the number of seconds within which an user can try to login. • <i>x</i>—Specifies the time period an user is banned if the user fails to successfully login.
Step 6	end Example: <pre>Device(config)# end</pre>	Exits global configuration mode and returns to privileged EXEC mode.
Step 7	show aaa local user blocked Example: <pre>Device# show aaa local user blocked</pre>	Displays the list of local users who were blocked.
Step 8	clear aaa local user blocked username <i>username</i> Example: <pre>Device# clear aaa local user blocked username user1</pre>	Clears the information about the blocked local user.

Example

The following is sample output from the **show aaa local user blocked** command:

```
Device# show aaa local user blocked

      Local-user              State
-----
      user1                   Watched (till 11:34:42 IST Feb 5 2015)
```

Configuring an Authenticator Switch with NEAT

Configuring this feature requires that one switch outside a wiring closet is configured as a supplicant and is connected to an authenticator switch.



Note

- The authenticator switch interface configuration must be restored to access mode by explicitly flapping it if a line card is removed and inserted in the chassis when CISP or NEAT session is active.
- The *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ISE, which sets the interface as a trunk after the supplicant is successfully authenticated.

Beginning in privileged EXEC mode, follow these steps to configure a switch as an authenticator:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**
3. **interface *interface-id***
4. **switchport mode access**
5. **authentication port-control auto**
6. **dot1x pae authenticator**
7. **spanning-tree portfast**
8. **end**
9. **show running-config interface *interface-id***
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	cisp enable Example: Switch(config)# cisp enable	Enables CISP.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 4	switchport mode access Example: Switch(config-if)# switchport mode access	Sets the port mode to access .
Step 5	authentication port-control auto Example: Switch(config-if)# authentication port-control auto	Sets the port-authentication mode to auto .

	Command or Action	Purpose
Step 6	dot1x pae authenticator Example: <pre>Switch(config-if)# dot1x pae authenticator</pre>	Configures the interface as a port access entity (PAE) authenticator.
Step 7	spanning-tree portfast Example: <pre>Switch(config-if)# spanning-tree portfast trunk</pre>	Enables Port Fast on an access port connected to a single workstation or server..
Step 8	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 9	show running-config interface <i>interface-id</i> Example: <pre>Switch# show running-config interface gigabitethernet 2/0/1</pre>	Verifies your configuration.
Step 10	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file. Note Saving changes to the configuration file will mean that the authenticator interface will continue to be in trunk mode after reload. If you want the authenticator interface to remain as an access port, do not save your changes to the configuration file.

Configuring a Supplicant Switch with NEAT

Beginning in privileged EXEC mode, follow these steps to configure a switch as a supplicant:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**
3. **dot1x credentials *profile***
4. **username *suppswitch***
5. **password *password***
6. **dot1x supplicant force-multicast**
7. **interface *interface-id***
8. **switchport trunk encapsulation dot1q**

9. `switchport mode trunk`
10. `dot1x pae supplicant`
11. `dot1x credentials profile-name`
12. `end`
13. `show running-config interface interface-id`
14. `copy running-config startup-config`
15. Configuring NEAT with Auto Smartports Macros

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	cisp enable Example: <pre>Switch(config)# cisp enable</pre>	Enables CISP.
Step 3	dot1x credentials profile Example: <pre>Switch(config)# dot1x credentials test</pre>	Creates 802.1x credentials profile. This must be attached to the port that is configured as supplicant.
Step 4	username suppswitch Example: <pre>Switch(config)# username suppswitch</pre>	Creates a username.
Step 5	password password Example: <pre>Switch(config)# password myswitch</pre>	Creates a password for the new username.
Step 6	dot1x supplicant force-multicast Example: <pre>Switch(config)# dot1x supplicant force-multicast</pre>	<p>Forces the switch to send only multicast EAPOL packets when it receives either unicast or multicast packets.</p> <p>This also allows NEAT to work on the supplicant switch in all host modes.</p>
Step 7	interface interface-id Example:	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Switch(config)# <code>interface gigabitethernet1/0/1</code>	
Step 8	switchport trunk encapsulation dot1q Example: Switch(config-if)# <code>switchport trunk encapsulation dot1q</code>	Sets the port to trunk mode.
Step 9	switchport mode trunk Example: Switch(config-if)# <code>switchport mode trunk</code>	Configures the interface as a VLAN trunk port.
Step 10	dot1x pae supplicant Example: Switch(config-if)# <code>dot1x pae supplicant</code>	Configures the interface as a port access entity (PAE) supplicant.
Step 11	dot1x credentials <i>profile-name</i> Example: Switch(config-if)# <code>dot1x credentials test</code>	Attaches the 802.1x credentials profile to the interface.
Step 12	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 13	show running-config interface <i>interface-id</i> Example: Switch# <code>show running-config interface gigabitethernet1/0/1</code>	Verifies your configuration.
Step 14	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.
Step 15	Configuring NEAT with Auto Smartports Macros	You can also use an Auto Smartports user-defined macro instead of the switch VSA to configure the authenticator

	Command or Action	Purpose
		switch. For more information, see the <i>Auto Smartports Configuration Guide</i> for this release.

Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs



Note You must configure a downloadable ACL on the ACS before downloading it to the switch.

After authentication on the port, you can use the **show ip access-list** privileged EXEC command to display the downloaded ACLs on the port.

Configuring Downloadable ACLs

The policies take effect after client authentication and the client IP address addition to the IP device tracking table. The switch then applies the downloadable ACL to the port.

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **ip device tracking**
3. **aaa new-model**
4. **aaa authorization network default local group radius**
5. **radius-server vsa send authentication**
6. **interface *interface-id***
7. **ip access-group *acl-id* in**
8. **show running-config interface *interface-id***
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	ip device tracking Example: <pre>Switch(config)# ip device tracking</pre>	Sets the ip device tracking table.

	Command or Action	Purpose
Step 3	aaa new-model Example: <pre>Switch(config)# aaa new-model</pre>	Enables AAA.
Step 4	aaa authorization network default local group radius Example: <pre>Switch(config)# aaa authorization network default local group radius</pre>	Sets the authorization method to local. To remove the authorization method, use the no aaa authorization network default local group radius command.
Step 5	radius-server vsa send authentication Example: <pre>Switch(config)# radius-server vsa send authentication</pre>	Configures the radius vsa send authentication.
Step 6	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet2/0/4</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 7	ip access-group <i>acl-id</i> in Example: <pre>Switch(config-if)# ip access-group default_acl in</pre>	Configures the default ACL on the port in the input direction. Note The <i>acl-id</i> is an access list name or number.
Step 8	show running-config interface <i>interface-id</i> Example: <pre>Switch(config-if)# show running-config interface gigabitethernet2/0/4</pre>	Verifies your configuration.
Step 9	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Downloadable Policy

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **access-list *access-list-number* { deny | permit } { hostname | any | host } log**
3. **interface *interface-id***
4. **ip access-group *acl-id* in**
5. **exit**
6. **aaa new-model**
7. **aaa authorization network default group radius**
8. **ip device tracking**
9. **ip device tracking probe [count | interval | use-svi]**
10. **radius-server vsa send authentication**
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> { deny permit } { hostname any host } log Example: <pre>Switch(config)# access-list 1 deny any log</pre>	<p>Defines the default port ACL.</p> <p>The access-list-number is a decimal number from 1 to 99 or 1300 to 1999.</p> <p>Enter deny or permit to specify whether to deny or permit access if conditions are matched.</p> <p>The source is the source address of the network or host that sends a packet, such as this:</p> <ul style="list-style-type: none"> • hostname: The 32-bit quantity in dotted-decimal format. • any: The keyword any as an abbreviation for source and source-wildcard value of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard value. • host: The keyword host as an abbreviation for source and source-wildcard of source 0.0.0.0. <p>(Optional) Applies the source-wildcard wildcard bits to the source.</p> <p>(Optional) Enters log to cause an informational logging message about the packet that matches the entry to be sent to the console.</p>
Step 3	interface <i>interface-id</i> Example:	Enters interface configuration mode.

	Command or Action	Purpose
	<code>Switch(config)# interface gigabitethernet2/0/2</code>	
Step 4	<p>ip access-group <i>acl-id</i> in</p> <p>Example:</p> <pre>Switch(config-if)# ip access-group default_acl in</pre>	<p>Configures the default ACL on the port in the input direction.</p> <p>Note The <i>acl-id</i> is an access list name or number.</p>
Step 5	<p>exit</p> <p>Example:</p> <pre>Switch(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	<p>aaa new-model</p> <p>Example:</p> <pre>Switch(config)# aaa new-model</pre>	Enables AAA.
Step 7	<p>aaa authorization network default group radius</p> <p>Example:</p> <pre>Switch(config)# aaa authorization network default group radius</pre>	Sets the authorization method to local. To remove the authorization method, use the no aaa authorization network default group radius command.
Step 8	<p>ip device tracking</p> <p>Example:</p> <pre>Switch(config)# ip device tracking</pre>	<p>Enables the IP device tracking table.</p> <p>To disable the IP device tracking table, use the no ip device tracking global configuration commands.</p>
Step 9	<p>ip device tracking probe [count interval use-svi]</p> <p>Example:</p> <pre>Switch(config)# ip device tracking probe count</pre>	<p>(Optional) Configures the IP device tracking table:</p> <ul style="list-style-type: none"> • count <i>count</i>—Sets the number of times that the switch sends the ARP probe. The range is from 1 to 5. The default is 3. • interval <i>interval</i>—Sets the number of seconds that the switch waits for a response before resending the ARP probe. The range is from 30 to 300 seconds. The default is 30 seconds. • use-svi—Uses the switch virtual interface (SVI) IP address as source of ARP probes.
Step 10	<p>radius-server vsa send authentication</p> <p>Example:</p>	Configures the network access server to recognize and use vendor-specific attributes.

	Command or Action	Purpose
	Switch(config)# <code>radius-server vsa send authentication</code>	Note The downloadable ACL must be operational.
Step 11	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.

Configuring VLAN ID-based MAC Authentication

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. `configure terminal`
2. `mab request format attribute 32 vlan access-vlan`
3. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	mab request format attribute 32 vlan access-vlan Example: Switch(config)# <code>mab request format attribute 32 vlan access-vlan</code>	Enables VLAN ID-based MAC authentication.
Step 3	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Flexible Authentication Ordering

The examples used in the instructions below changes the order of Flexible Authentication Ordering so that MAB is attempted before IEEE 802.1X authentication (dot1x). MAB is configured as the first authentication method, so MAB will have priority over all other authentication methods.

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **switchport mode access**
4. **authentication order [dot1x | mab] | {webauth}**
5. **authentication priority [dot1x | mab] | {webauth}**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Switch(config-if)# switchport mode access	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	authentication order [dot1x mab] {webauth} Example: Switch(config-if)# authentication order mab dot1x	(Optional) Sets the order of authentication methods used on a port.
Step 5	authentication priority [dot1x mab] {webauth} Example: Switch(config-if)# authentication priority mab dot1x	(Optional) Adds an authentication method to the port-priority list.

	Command or Action	Purpose
Step 6	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring Open1x

Beginning in privileged EXEC mode, follow these steps to enable manual control of the port authorization state:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication control-direction** {both | in}
5. **authentication fallback** *name*
6. **authentication host-mode** [multi-auth | multi-domain | multi-host | single-host]
7. **authentication open**
8. **authentication order** [dot1x | mab] | {webauth}
9. **authentication periodic**
10. **authentication port-control** {auto | force-authorized | force-un authorized}
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: <pre>Switch(config-if)# switchport mode access</pre>	Sets the port to access mode only if you configured the RADIUS server.

	Command or Action	Purpose
Step 4	authentication control-direction {both in} Example: <pre>Switch(config-if)# authentication control-direction both</pre>	(Optional) Configures the port control as unidirectional or bidirectional.
Step 5	authentication fallback <i>name</i> Example: <pre>Switch(config-if)# authentication fallback profile1</pre>	(Optional) Configures a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
Step 6	authentication host-mode [multi-auth multi-domain multi-host single-host] Example: <pre>Switch(config-if)# authentication host-mode multi-auth</pre>	(Optional) Sets the authorization manager mode on a port.
Step 7	authentication open Example: <pre>Switch(config-if)# authentication open</pre>	(Optional) Enables or disable open access on a port.
Step 8	authentication order [dot1x mab] {webauth} Example: <pre>Switch(config-if)# authentication order dot1x webauth</pre>	(Optional) Sets the order of authentication methods used on a port.
Step 9	authentication periodic Example: <pre>Switch(config-if)# authentication periodic</pre>	(Optional) Enables or disable reauthentication on a port.
Step 10	authentication port-control {auto force-authorized force-un authorized} Example: <pre>Switch(config-if)# authentication port-control auto</pre>	(Optional) Enables manual control of the port authorization state.

	Command or Action	Purpose
Step 11	end Example: <pre>Switch(config-if) # end</pre>	Returns to privileged EXEC mode.

Disabling 802.1x Authentication on the Port

You can disable 802.1x authentication on the port by using the **no dot1x pae** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to disable 802.1x authentication on the port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **no dot1x pae authenticator**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 2/0/1</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: <pre>Switch(config-if)# switchport mode access</pre>	(Optional) Sets the port to access mode only if you configured the RADIUS server.
Step 4	no dot1x pae authenticator Example: <pre>Switch(config-if)# no dot1x pae authenticator</pre>	Disables 802.1x authentication on the port.

	Command or Action	Purpose
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Resetting the 802.1x Authentication Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the 802.1x authentication configuration to the default values. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **dot1x default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/2</pre>	Enters interface configuration mode, and specify the port to be configured.
Step 3	dot1x default Example: <pre>Switch(config-if)# dot1x default</pre>	Resets the 802.1x parameters to the default values.
Step 4	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.

Monitoring 802.1x Statistics and Status

Table 144: Privileged EXEC show Commands

Command	Purpose
show dot1x all statistics	Displays 802.1x statistics for all ports
show dot1x interface <i>interface-id</i> statistics	Displays 802.1x statistics for a specific port
show dot1x all [count details statistics summary]	Displays the 802.1x administrative and operational status for a switch
show dot1x interface <i>interface-id</i>	Displays the 802.1x administrative and operational status for a specific port

Table 145: Global Configuration Commands

Command	Purpose
no dot1x logging verbose	Filters verbose 802.1x authentication messages (beginning with Cisco IOS Release 12.2(55)SE)

For detailed information about the fields in these displays, see the command reference for this release.



CHAPTER 69

Configuring MACsec Encryption

- [Finding Feature Information, on page 1413](#)
- [Information About MACsec Encryption, on page 1413](#)
- [Configuring MKA and MACsec, on page 1420](#)
- [Configuring MACsec MKA using PSK, on page 1423](#)
- [Configuring MACsec MKA using EAP-TLS, on page 1425](#)
- [Configuring Cisco TrustSec MACsec, on page 1439](#)
- [Configuration Examples for Configuring MACsec Encryption, on page 1444](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

Information About MACsec Encryption

This chapter describes how to configure Media Access Control Security (MACsec) encryption on the Catalyst switches. MACsec is the IEEE 802.1AE standard for authenticating and encrypting packets between two MACsec-capable devices. The switch also supports MACsec link layer switch-to-switch security by using Cisco TrustSec Network Device Admission Control (NDAC) and the Security Association Protocol (SAP) key exchange. Link layer security can include both packet authentication between switches and MACsec encryption between switches (encryption is optional).



Note MACsec is not supported on switches running the NPE or the LAN base image.

All downlink ports on the switch can run Cisco TrustSec MACsec link layer switch-to-switch security.

Table 146: MACsec Support on Switch Ports

Interface	Connections	MACsec support
Switchports connected to other switches	Switch-to-switch	Cisco TrustSec NDAC MACsec

Cisco TrustSec and Cisco SAP are meant only for switch-to-switch links and are not supported on switch ports connected to end hosts, such as PCs or IP phones. Cisco NDAC and SAP are mutually exclusive with Network Edge Access Topology (NEAT), which is used for compact switches to extend security outside the wiring closet.

Media Access Control Security and MACsec Key Agreement

MACsec, defined in 802.1AE, provides MAC-layer encryption over wired networks by using out-of-band methods for encryption keying. The MACsec Key Agreement (MKA) Protocol provides the required session keys and manages the required encryption keys. MKA and MACsec are implemented after successful authentication using the 802.1x Extensible Authentication Protocol (EAP-TLS) or Pre Shared Key (PSK) framework.

A switch using MACsec accepts either MACsec or non-MACsec frames, depending on the policy associated with the MKA peer. MACsec frames are encrypted and protected with an integrity check value (ICV). When the switch receives frames from the MKA peer, it decrypts them and calculates the correct ICV by using session keys provided by MKA. The switch compares that ICV to the ICV within the frame. If they are not identical, the frame is dropped. The switch also encrypts and adds an ICV to any frames sent over the secured port (the access point used to provide the secure MAC service to a MKA peer) using the current session key.

The MKA Protocol manages the encryption keys used by the underlying MACsec protocol. The basic requirements of MKA are defined in 802.1x-REV. The MKA Protocol extends 802.1x to allow peer discovery with confirmation of mutual authentication and sharing of MACsec secret keys to protect data exchanged by the peers.

The EAP framework implements MKA as a newly defined EAP-over-LAN (EAPoL) packet. EAP authentication produces a master session key (MSK) shared by both partners in the data exchange. Entering the EAP session ID generates a secure connectivity association key name (CKN). The switch acts as the authenticator for both uplink and downlink; and acts as the key server for downlink. It generates a random secure association key (SAK), which is sent to the client partner. The client is never a key server and can only interact with a single MKA entity, the key server. After key derivation and generation, the switch sends periodic transports to the partner at a default interval of 2 seconds.

The packet body in an EAPoL Protocol Data Unit (PDU) is referred to as a MACsec Key Agreement PDU (MKPDU). MKA sessions and participants are deleted when the MKA lifetime (6 seconds) passes with no MKPDU received from a participant. For example, if a MKA peer disconnects, the participant on the switch continues to operate MKA until 6 seconds have elapsed after the last MKPDU is received from the MKA peer.



Note Integrity check value (ICV) indicator in MKPDU is optional. ICV is not optional when the traffic is encrypted.

EAPoL Announcements indicate the use of the type of keying material. The announcements can be used to announce the capability of the supplicant as well as the authenticator. Based on the capability of each side, the largest common denominator of the keying material could be used.

Prior to Cisco IOS XE Fuji 16.8.1a, should-secure was supported for MKA and SAP. With should-secure enabled, if the peer is configured for MACsec, the data traffic is encrypted, otherwise it is sent in clear text. Starting with Cisco IOS XE Fuji 16.8.1a, must-secure support is enabled on both the ingress and the egress. Must-secure is supported for MKA and SAP. With must-secure enabled, only EAPoL traffic will not be encrypted. The rest of the traffic will be encrypted. Unencrypted packets are dropped.



Note Must-secure mode is enabled by default.

MKA Policies

To enable MKA on an interface, a defined MKA policy should be applied to the interface. Removing the MKA policy disables MKA on that interface. You can configure these options:

- Policy name, not to exceed 16 ASCII characters.
- Confidentiality (encryption) offset of 0, 30, or 50 bytes for each physical interface
- Replay protection. You can configure MACsec window size, as defined by the number of out-of-order frames that are accepted. This value is used while installing the security associations in the MACsec. A value of 0 means that frames are accepted only in the correct order.

Virtual Ports

You use virtual ports for multiple secured connectivity associations on a single physical port. Each connectivity association (pair) represents a virtual port, with a maximum of two virtual ports per physical port. Only one of the two virtual ports can be part of a data VLAN; the other must externally tag its packets for the voice VLAN. You cannot simultaneously host secured and unsecured sessions in the same VLAN on the same port. Because of this limitation, 802.1x multiple authentication mode is not supported.

The exception to this limitation is in multiple-host mode when the first MACsec supplicant is successfully authenticated and connected to a hub that is connected to the switch. A non-MACsec host connected to the hub can send traffic without authentication because it is in multiple-host mode. We do not recommend using multi-host mode because after the first successful client, authentication is not required for other clients.

Virtual ports represent an arbitrary identifier for a connectivity association and have no meaning outside the MKA Protocol. A virtual port corresponds to a separate logical port ID. Valid port IDs for a virtual port are 0x0002 to 0xFFFF. Each virtual port receives a unique secure channel identifier (SCI) based on the MAC address of the physical interface concatenated with a 16-bit port ID.

MACsec and Stacking

A (Catalyst 3560cx) Switch stack master running MACsec maintains the configuration files that show which ports on a member switch support MACsec. The stack master performs these functions:

- Processes secure channel and secure association creation and deletion
- Sends secure association service requests to the stack members.
- Processes packet number and replay-window information from local or remote ports and notifies the key management protocol.
- Sends MACsec initialization requests with the globally configured options to new switches that are added to the stack.
- Sends any per-port configuration to the member switches.

A member switch performs these functions:

- Processes MACsec initialization requests from the stack master.
- Processes MACsec service requests sent by the stack master.
- Sends information about local ports to the stack master.

In case of a stack master changeover, all secured sessions are brought down and then reestablished. The authentication manager recognizes any secured sessions and initiates teardown of these sessions.



Note If you are using 1G SFP modules for inter switch connection, change system MTU to 1550 byte to ensure support of MACsec overhead.

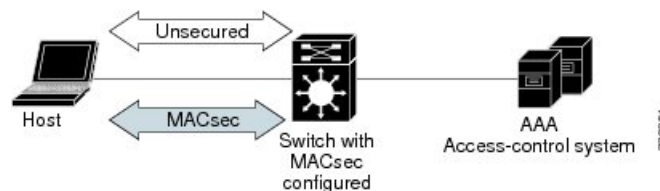
MACsec, MKA and 802.1x Host Modes

You can use MACsec and the MKA Protocol with 802.1x single-host mode or Multi Domain Authentication (MDA) mode. Multiple authentication mode is not supported.

Single-Host Mode

The figure shows how a single EAP authenticated session is secured by MACsec by using MKA

Figure 113: MACsec in Single-Host Mode with a Secured Data Session



MKA Statistics

Some MKA counters are aggregated globally, while others are updated both globally and per session. You can also obtain information about the status of MKA sessions.

This is an example of the `show mka statistics` command output:

```
Switch# show mka statistics
MKA Global Statistics
=====
MKA Session Totals
Secured..... 32
Reauthentication Attempts.. 31

Deleted (Secured)..... 1
Keepalive Timeouts..... 0

CA Statistics
Pairwise CAKs Derived..... 32
Pairwise CAK Rekeys..... 31
Group CAKs Generated..... 0
Group CAKs Received..... 0

SA Statistics
```



```

SAKs Generated..... 32
SAKs Rekeyed..... 31
SAKs Received..... 0
SAK Responses Received..... 32

MKPDU Statistics
MKPDUs Validated & Rx..... 580
"Distributed SAK"..... 0
"Distributed CAK"..... 0
MKPDUs Transmitted..... 597
"Distributed SAK"..... 32
"Distributed CAK"..... 0

MKA Error Counter Totals
=====
Bring-up Failures..... 0
Reauthentication Failures..... 0

SAK Failures
SAK Generation..... 0
Hash Key Generation..... 0
SAK Encryption/Wrap..... 0
SAK Decryption/Unwrap..... 0

CA Failures
Group CAK Generation..... 0
Group CAK Encryption/Wrap..... 0
Group CAK Decryption/Unwrap..... 0
Pairwise CAK Derivation..... 0
CKN Derivation..... 0
ICK Derivation..... 0
KEK Derivation..... 0
Invalid Peer MACsec Capability.. 2

MACsec Failures
Rx SC Creation..... 0
Tx SC Creation..... 0
Rx SA Installation..... 0
Tx SA Installation..... 0

MKPDU Failures
MKPDU Tx..... 0
MKPDU Rx Validation..... 0
MKPDU Rx Bad Peer MN..... 0
MKPDU Rx Non-recent Peerlist MN.. 0

```

Information About MACsec MKA using EAP-TLS

MACsec MKA is supported on switch-to-switch links. Using IEE 802.1X Port-based Authentication with Extensible Authentication Protocol (EAP-TLS), you can configure MACsec MKA between device uplink ports. EAP-TLS allows mutual authentication and obtains an MSK (master session key) from which the connectivity association key (CAK) is derived for MKA operations. Device certificates are carried, using EAP-TLS, for authentication to the AAA server.

Prerequisites for MACsec MKA using EAP-TLS

- Ensure that you have a Certificate Authority (CA) server configured for your network.
- Generate a CA certificate.

- Ensure that you have configured Cisco Identity Services Engine (ISE) Release 2.0.
- Ensure that both the participating devices, the CA server, and Cisco Identity Services Engine (ISE) are synchronized using Network Time Protocol (NTP). If time is not synchronized on all your devices, certificates will not be validated.
- Ensure that 802.1x authentication and AAA are configured on your device.

Limitations for MACsec MKA using EAP-TLS

- MKA is not supported on port-channels.
- Cisco Catalyst 3560-CX Switches do not support MACSec MKA configuration on EtherChannels.
- MKA is not supported with High Availability and local authentication.
- MKA/EAPTLS is not supported for promiscuous PVLAN Primary port.
- While configuring MACsec MKA using EAP-TLS, MACsec secure channels encrypt counters does not increment before first Rekey.

Cisco TrustSec Overview

The table below lists the TrustSec features to be eventually implemented on TrustSec-enabled Cisco switches. Successive general availability releases of TrustSec will expand the number of switches supported and the number of TrustSec features supported per switch.

Cisco TrustSec Feature	Description
802.1AE Tagging (MACsec)	<p>Protocol for IEEE 802.1AE-based wire-rate hop-to-hop Layer 2 encryption.</p> <p>Between MACsec-capable devices, packets are encrypted on egress from the transmitting device, decrypted on ingress to the receiving device, and in the clear within the devices.</p> <p>This feature is only available between TrustSec hardware-capable devices.</p>
Endpoint Admission Control (EAC)	<p>EAC is an authentication process for an endpoint user or a device connecting to the TrustSec domain. Usually EAC takes place at the access level switch. Successful authentication and authorization in the EAC process results in Security Group Tag assignment for the user or device. Currently EAC can be 802.1X, MAC Authentication Bypass (MAB), and Web Authentication Proxy (WebAuth).</p>

Cisco TrustSec Feature	Description
Network Device Admission Control (NDAC)	NDAC is an authentication process where each network device in the TrustSec domain can verify the credentials and trustworthiness of its peer device. NDAC utilizes an authentication framework based on IEEE 802.1X port-based authentication and uses EAP-FAST as its EAP method. Successful authentication and authorization in NDAC process results in Security Association Protocol negotiation for IEEE 802.1AE encryption.
Security Association Protocol (SAP)	After NDAC authentication, the Security Association Protocol (SAP) automatically negotiates keys and the cipher suite for subsequent MACSec link encryption between TrustSec peers. SAP is defined in IEEE 802.11i.
Security Group Tag (SGT)	An SGT is a 16-bit single label indicating the security classification of a source in the TrustSec domain. It is appended to an Ethernet frame or an IP packet.
SGT Exchange Protocol (SXP)	Security Group Tag Exchange Protocol (SXP). With SXP, devices that are not TrustSec-hardware-capable can receive SGT attributes for authenticated users and devices from the Cisco Identity Services Engine (ISE) or the Cisco Secure Access Control System (ACS). The devices can then forward a sourceIP-to-SGT binding to a TrustSec-hardware-capable device will tag the source traffic for SGACL enforcement.

When both ends of a link support 802.1AE MACsec, SAP negotiation occurs. An EAPOL-key exchange occurs between the supplicant and the authenticator to negotiate a cipher suite, exchange security parameters, and manage keys. Successful completion of these tasks results in the establishment of a security association (SA).

Depending on your software version and licensing and link hardware support, SAP negotiation can use one of these modes of operation:

- Galois Counter Mode (GCM)—authentication and encryption
- GCM authentication (GMAC)— GCM authentication, no encryption
- No Encapsulation—no encapsulation (clear text)
- Null—encapsulation, no authentication or encryption

Configuring MKA and MACsec

Default MACsec MKA Configuration

MACsec is disabled. No MKA policies are configured.

Configuring an MKA Policy

SUMMARY STEPS

1. **configure terminal**
2. **mka policy** *policy name*
3. **confidentiality-offset** *Offset value*
4. **replay-protection window-size** *frames*
5. **end**
6. **show mka policy**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	mka policy <i>policy name</i>	Identify an MKA policy, and enter MKA policy configuration mode. The maximum policy name length is 16 characters.
Step 3	confidentiality-offset <i>Offset value</i>	Set the Confidentiality (encryption) offset for each physical interface Note Offset Value can be 0, 30 or 50. If you are using Anyconnect on the client, it is recommended to use Offset 0.
Step 4	replay-protection window-size <i>frames</i>	Enable replay protection, and configure the window size in number of frames. The range is from 0 to 4294967295. The default window size is 0. Entering a window size of 0 is not the same as entering the no replay-protection command . Configuring a window size of 0 uses replay protection with a strict ordering of frames. Entering no replay-protection turns off MACsec replay-protection.
Step 5	end	Return to privileged EXEC mode.
Step 6	show mka policy	Verify your entries.

Example

This example configures the MKA policy *relay-policy*:

```
Switch(config)# mka policy replay-policy
Switch(config-mka-policy)# confidentiality-offset 0
Switch(config-mka-policy)# replay-protection window-size 300
Switch(config-mka-policy)# end
```

Configuring MACsec on an Interface

Follow these steps to configure MACsec on an interface with one MACsec session for voice and one for data:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport access vlan***vlan-id*
5. **switchport mode access**
6. **macsec**
7. **authentication event linksec fail action authorize vlan** *vlan-id*
8. **authentication host-mode multi-domain**
9. **authentication linksec policy must-secure**
10. **authentication port-control auto**
11. **authentication periodic**
12. **authentication timer reauthenticate**
13. **authentication violation protect**
14. **mka policy** *policy name*
15. **dot1x pae authenticator**
16. **spanning-tree portfast**
17. **end**
18. **show authentication session interface** *interface-id*
19. **show authentication session interface** *interface-id* details
20. **show macsec interface** *interface-id*
21. **show mka sessions**
22. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Identify the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 4	switchport access vlan <i>vlan-id</i>	Configure the access VLAN for the port.
Step 5	switchport mode access	Configure the interface as an access port.
Step 6	macsec	Enable 802.1ae MACsec on the interface.
Step 7	authentication event linksec fail action authorize vlan <i>vlan-id</i>	(Optional) Specify that the switch processes authentication link-security failures resulting from unrecognized user credentials by authorizing a restricted VLAN on the port after a failed authentication attempt.
Step 8	authentication host-mode multi-domain	Configure authentication manager mode on the port to allow both a host and a voice device to be authenticated on the 802.1x-authorized port. If not configured, the default host mode is single.
Step 9	authentication linksec policy must-secure	Set the LinkSec security policy to secure the session with MACsec if the peer is available. If not set, the default is <i>should secure</i> .
Step 10	authentication port-control auto	Enable 802.1x authentication on the port. The port changes to the authorized or unauthorized state based on the authentication exchange between the switch and the client.
Step 11	authentication periodic	Enable or Disable Reauthentication for this port .
Step 12	authentication timer reauthenticate	Enter a value between 1 and 65535. Obtains re-authentication timeout value from the server.
Step 13	authentication violation protect	Configure the port to drop unexpected incoming MAC addresses when a new device connects to a port or when a device connects to a port after the maximum number of devices are connected to that port. If not configured, the default is to shut down the port.
Step 14	mka policy <i>policy name</i>	Apply an existing MKA protocol policy to the interface, and enable MKA on the interface. If no MKA policy was configured (by entering the mka policy global configuration command), you must apply the MKA default policy to the interface by entering the mka default-policy interface configuration command

	Command or Action	Purpose
Step 15	dot1x pae authenticator	Configure the port as an 802.1x port access entity (PAE) authenticator.
Step 16	spanning-tree portfast	Enable spanning tree Port Fast on the interface in all its associated VLANs. When Port Fast feature is enabled, the interface changes directly from a blocking state to a forwarding state without making the intermediate spanning-tree state changes
Step 17	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 18	show authentication session interface <i>interface-id</i>	Verify the authorized session security status.
Step 19	show authentication session interface <i>interface-id</i> details	Verify the details of the security status of the authorized session.
Step 20	show macsec interface <i>interface-id</i>	Verify MacSec status on the interface.
Step 21	show mka sessions	Verify the established mka sessions.
Step 22	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring MACsec MKA using PSK

SUMMARY STEPS

1. **configure terminal**
2. **key chain *key-chain-name* macsec**
3. **key *hex-string***
4. **key-string { [0/6/7] *pwd-string* | *pwd-string* }**
5. **lifetime local [start timestamp {*hh::mm::ss* | *day* | *month* | *year*}] [duration *seconds* | *end timestamp* {*hh::mm::ss* | *day* | *month* | *year*}]**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.

	Command or Action	Purpose
Step 2	key chain <i>key-chain-name</i> macsec	Configures a key chain and enters the key chain configuration mode.
Step 3	key <i>hex-string</i>	Configures a unique identifier for each key in the keychain and enters the keychain's key configuration mode. Note For 128-bit encryption, use 32 hex digit key-string.
Step 4	key-string { [0/6/7] <i>pwd-string</i> <i>pwd-string</i> }	Sets the password for a key string. Only hex characters must be entered.
Step 5	lifetime local [<i>start timestamp {hh::mm::ss day month year}</i>] [duration <i>seconds</i> <i>end timestamp {hh::mm::ss day month year}</i>]	Sets the lifetime of the pre shared key.
Step 6	end	Returns to privileged EXEC mode.

Example

Following is an indicative example:

```
Switch(config)# Key chain keychain1 macsec
Switch(config-key-chain)# key 1000
Switch(config-keychain-key)# cryptographic-algorithm gcm-aes-128
Switch(config-keychain-key)# key-string 12345678901234567890123456789012
Switch(config-keychain-key)# lifetime local 12:12:00 July 28 2016 12:19:00 July
28 2016
Switch(config-keychain-key)# end
```

Configuring MACsec MKA on an Interface using PSK

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **macsec network-link**
4. **mka policy** *policy-name*
5. **mka pre-shared-key** *key-chain key-chain name*
6. **macsec replay-protection window-size** *frame number*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<code>interface <i>interface-id</i></code>	Enters interface configuration mode.
Step 3	<code>macsec network-link</code>	Enables MACsec on the interface.
Step 4	<code>mka policy <i>policy-name</i></code>	Configures an MKA policy.
Step 5	<code>mka pre-shared-key key-chain <i>key-chain name</i></code>	Configures an MKA pre-shared-key key-chain name. Note The MKA pre-shared key can be configured on either physical interface or sub-interfaces and not on both.
Step 6	<code>macsec replay-protection window-size <i>frame number</i></code>	Sets the MACsec window size for replay protection.
Step 7	<code>end</code>	Returns to privileged EXEC mode.

Example

Following is an indicative example:

```
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# mka policy mka_policy
Switch(config-if)# mka pre-shared-key key-chain key-chain-name
Switch(config-if)# macsec replay-protection window-size 10
Switch(config-if)# end
```

What to do next

It is not recommended to change the MKA policy on an interface with MKA PSK configured when the session is running. However, if a change is required, you must reconfigure the policy as follows:

1. Disable the existing session by removing `macsec network-link` configuration on each of the participating node using the **no macsec network-link** command
2. Configure the MKA policy on the interface on each of the participating node using the **mka policy policy-name** command.
3. Enable the new session on each of the participating node by using the **macsec network-link** command.

Configuring MACsec MKA using EAP-TLS

To configure MACsec with MKA on point-to-point links, perform these tasks:

- Configure Certificate Enrollment
 - Generate Key Pairs
 - Configure SCEP Enrollment
 - Configure Certificates Manually

- Configure an Authentication Policy
- Configure EAP-TLS Profiles and IEEE 802.1x Credentials
- Configure MKA MACsec using EAP-TLS on Interfaces

Remote Authentication

Generating Key Pairs

Procedure

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>crypto key generate rsa label <i>label-name</i> general-keys modulus <i>size</i></code>	Generates a RSA key pair for signing and encryption. You can also assign a label to each key pair using the label keyword. The label is referenced by the trustpoint that uses the key pair. If you do not assign a label, the key pair is automatically labeled <Default-RSA-Key>. If you do not use additional keywords this command generates one general purpose RSA key pair. If the modulus is not specified, the default key modulus of 1024 is used. You can specify other modulus sizes with the modulus keyword.
Step 3	<code>end</code>	Returns to privileged EXEC mode.
Step 4	<code>show authentication session interface <i>interface-id</i></code>	Verifies the authorized session security status.
Step 5	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Enrollment using SCEP

Simple Certificate Enrollment Protocol (SCEP) is a Cisco-developed enrollment protocol that uses HTTP to communicate with the certificate authority (CA) or registration authority (RA). SCEP is the most commonly used method for sending and receiving requests and certificates.

Procedure

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>crypto pki trustpoint <i>server name</i></code>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 3	<code>enrollment url <i>url name pem</i></code>	Specifies the URL of the CA on which your device should send certificate requests.

	Command or Action	Purpose
		An IPv6 address can be added in the URL enclosed in brackets. For example: http:// [2001:DB8:1:1::1]:80. The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.
Step 4	rsa keypair <i>label</i>	Specifies which key pair to associate with the certificate. Note The rsa keypair name must match the trust-point name.
Step 5	serial-number none	The none keyword specifies that a serial number will not be included in the certificate request.
Step 6	ip-address none	The none keyword specifies that no IP address should be included in the certificate request.
Step 7	revocation-check <i>crl</i>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 8	auto-enroll <i>percent regenerate</i>	Enables auto-enrollment, allowing the client to automatically request a rollover certificate from the CA. If auto-enrollment is not enabled, the client must be manually re-enrolled in your PKI upon certificate expiration. By default, only the Domain Name System (DNS) name of the device is included in the certificate. Use the percent argument to specify that a new certificate will be requested after the percentage of the lifetime of the current certificate is reached. Use the regenerate keyword to generate a new key for the certificate even if a named key already exists. If the key pair being rolled over is exportable, the new key pair will also be exportable. The following comment will appear in the trustpoint configuration to indicate whether the key pair is exportable: “! RSA key pair associated with trustpoint is exportable.” It is recommended that a new key pair be generated for security reasons.
Step 9	crypto pki authenticate <i>name</i>	Retrieves the CA certificate and authenticates it.
Step 10	exit	Exits global configuration mode.
Step 11	show crypto pki certificate <i>trustpoint name</i>	Displays information about the certificate for the trust point.

Configuring Enrollment Manually

If your CA does not support SCEP or if a network connection between the router and CA is not possible. Perform the following task to set up manual certificate enrollment:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 3	enrollment url <i>url name pem</i>	Specifies the URL of the CA on which your device should send certificate requests. An IPv6 address can be added in the URL enclosed in brackets. For example: http:// [2001:DB8:1:1::1]:80. The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.
Step 4	rsa keypair <i>label</i>	Specifies which key pair to associate with the certificate.
Step 5	serial-number none	The none keyword specifies that a serial number will not be included in the certificate request.
Step 6	ip-address none	The none keyword specifies that no IP address should be included in the certificate request.
Step 7	revocation-check <i>crl</i>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 8	exit	Exits Global Configuration mode.
Step 9	crypto pki authenticate <i>name</i>	Retrieves the CA certificate and authenticates it.
Step 10	crypto pki enroll <i>name</i>	Generates certificate request and displays the request for copying and pasting into the certificate server. Enter enrollment information when you are prompted. For example, specify whether to include the device FQDN and IP address in the certificate request. You are also given the choice about displaying the certificate request to the console terminal. The base-64 encoded certificate with or without PEM headers as requested is displayed.
Step 11	crypto pki import <i>name certificate</i>	Imports a certificate via TFTP at the console terminal, which retrieves the granted certificate. The device attempts to retrieve the granted certificate via TFTP using the same filename used to send the request, except the extension is changed from “.req” to “.crt”. For

	Command or Action	Purpose
		usage key certificates, the extensions “-sign.crt” and “-encr.crt” are used. The device parses the received files, verifies the certificates, and inserts the certificates into the internal certificate database on the switch. Note Some CAs ignore the usage key information in the certificate request and issue general purpose usage certificates. If your CA ignores the usage key information in the certificate request, only import the general purpose certificate. The router will not use one of the two key pairs generated.
Step 12	<code>exit</code>	Exits global configuration mode.
Step 13	<code>show crypto pki certificate <i>trustpoint name</i></code>	Displays information about the certificate for the trust point.
Step 14	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling 802.1x Authentication and Configuring AAA

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>aaa new-model</code>	Enables AAA.
Step 4	<code>dot1x system-auth-control</code>	Enables 802.1X on your device.
Step 5	<code>radius server <i>name</i></code>	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode.
Step 6	<code>address <i>ip-address</i> auth-port <i>port-number</i> acct-port <i>port-number</i></code>	Configures the IPv4 address for the RADIUS server accounting and authentication parameters.
Step 7	<code>automate-tester username <i>username</i></code>	Enables the automated testing feature for the RADIUS server. With this practice, the device sends periodic test authentication messages to the RADIUS server. It looks for a RADIUS response from the server. A success

	Command or Action	Purpose
		message is not necessary - a failed authentication suffices, because it shows that the server is alive.
Step 8	<code>key string</code>	Configures the authentication and encryption key for all RADIUS communications between the device and the RADIUS server.
Step 9	<code>radius-server deadtime minutes</code>	Improves RADIUS response time when some servers might be unavailable and skips unavailable servers immediately.
Step 10	<code>exit</code>	Returns to global configuration mode.
Step 11	<code>aaa group server radius group-name</code>	Groups different RADIUS server hosts into distinct lists and distinct methods, and enters server group configuration mode.
Step 12	<code>server name</code>	Assigns the RADIUS server name.
Step 13	<code>exit</code>	Returns to global configuration mode.
Step 14	<code>aaa authentication dot1x default group group-name</code>	Sets the default authentication server group for IEEE 802.1x.
Step 15	<code>aaa authorization network default group group-name</code>	Sets the network authorization default group.

Configuring EAP-TLS Profile and 802.1x Credentials

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>eap profile profile-name</code>	Configures EAP profile and enters EAP profile configuration mode.
Step 4	<code>method tls</code>	Enables EAP-TLS method on the device.
Step 5	<code>pki-trustpoint name</code>	Sets the default PKI trustpoint.
Step 6	<code>exit</code>	Returns to global configuration mode.
Step 7	<code>dot1x credentials profile-name</code>	Configures 802.1x credentials profile and enters dot1x credentials configuration mode.
Step 8	<code>username username</code>	Sets the authentication user ID.
Step 9	<code>pki-trustpoint name</code>	Sets the default PKI trustpoint.

	Command or Action	Purpose
Step 10	end	Returns to privileged EXEC mode.

Applying the 802.1x MACsec MKA Configuration on Interfaces

To apply MACsec MKA using EAP-TLS to interfaces, perform the following task:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i>	Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 3	macsec network-link	Enables MACsec on the interface.
Step 4	authentication periodic	Enables reauthentication for this port.
Step 5	authentication timer reauthenticate interval	Sets the reauthentication interval.
Step 6	access-session host-mode multi-domain	Allows hosts to gain access to the interface.
Step 7	access-session closed	Prevents preauthentication access on the interface.
Step 8	access-session port-control auto	Sets the authorization state of a port.
Step 9	dot1x pae both	Configures the port as an 802.1X port access entity (PAE) supplicant and authenticator.
Step 10	dot1x credentials profile	Assigns a 802.1x credentials profile to the interface.
Step 11	dot1x supplicant eap profile <i>name</i>	Assigns the EAP-TLS profile to the interface.
Step 12	service-policy type control subscriber <i>control-policy name</i>	Applies a subscriber control policy to the interface.
Step 13	exit	Returns to privileged EXEC mode.
Step 14	show macsec interface	Displays MACsec details for the interface.
Step 15	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Local Authentication

Configuring the EAP Credentials using Local Authentication

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	aaa new-model	Enables AAA.
Step 4	aaa local authentication default authorization default	Sets the default local authentication and default local authorization method.
Step 5	aaa authentication dot1x default local	Sets the default local username authentication list for IEEE 802.1x.
Step 6	aaa authorization network default local	Sets an authorization method list for local user.
Step 7	aaa authorization credential-download default local	Sets an authorization method list for use of local credentials.
Step 8	exit	Returns to privileged EXEC mode.

Configuring the Local EAP-TLS Authentication and Authorization Profile

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	aaa new-model	Enables AAA.
Step 4	dot1x credentials <i>profile-name</i>	Configures the dot1x credentials profile and enters dot1x credentials configuration mode.
Step 5	username <i>name</i> password <i>password</i>	Sets the authentication user ID and password.
Step 6	exit	Returns to global configuration mode.
Step 7	aaa attribute list <i>list-name</i>	(Optional) Sets the AAA attribute list definition and enters attribute list configuration mode.
Step 8	aaa attribute type linksec-policy must-secure	(Optional) Specifies the AAA attribute type.

	Command or Action	Purpose
Step 9	<code>exit</code>	Returns to global configuration mode.
Step 10	<code>username name aaa attribute list name</code>	(Optional) Specifies the AAA attribute list for the user ID.
Step 11	<code>end</code>	Returns to privileged EXEC mode.

Configuring Enrollment using SCEP

Simple Certificate Enrollment Protocol (SCEP) is a Cisco-developed enrollment protocol that uses HTTP to communicate with the certificate authority (CA) or registration authority (RA). SCEP is the most commonly used method for sending and receiving requests and certificates.

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>crypto pki trustpoint server name</code>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 4	<code>enrollment url url name pem</code>	Specifies the URL of the CA on which your device should send certificate requests. An IPv6 address can be added in the URL enclosed in brackets. For example: <code>http:// [2001:DB8:1:1::1]:80</code> . The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.
Step 5	<code>rsa keypair label</code>	Specifies which key pair to associate with the certificate. Note The rsa keypair name must match the trust-point name.
Step 6	<code>serial-number none</code>	The none keyword specifies that a serial number will not be included in the certificate request.
Step 7	<code>ip-address none</code>	The none keyword specifies that no IP address should be included in the certificate request.
Step 8	<code>revocation-check crl</code>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 9	<code>auto-enroll percent regenerate</code>	Enables auto-enrollment, allowing the client to automatically request a rollover certificate from the CA.

	Command or Action	Purpose
		<p>If auto-enrollment is not enabled, the client must be manually re-enrolled in your PKI upon certificate expiration.</p> <p>By default, only the Domain Name System (DNS) name of the device is included in the certificate.</p> <p>Use the percent argument to specify that a new certificate will be requested after the percentage of the lifetime of the current certificate is reached.</p> <p>Use the regenerate keyword to generate a new key for the certificate even if a named key already exists.</p> <p>If the key pair being rolled over is exportable, the new key pair will also be exportable. The following comment will appear in the trustpoint configuration to indicate whether the key pair is exportable: “! RSA key pair associated with trustpoint is exportable.”</p> <p>It is recommended that a new key pair be generated for security reasons.</p>
Step 10	crypto pki authenticate <i>name</i>	Retrieves the CA certificate and authenticates it.
Step 11	exit	Exits global configuration mode.
Step 12	show crypto pki certificate <i>trustpoint name</i>	Displays information about the certificate for the trust point.

Configuring Enrollment Manually

If your CA does not support SCEP or if a network connection between the router and CA is not possible. Perform the following task to set up manual certificate enrollment:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 4	enrollment url <i>url name pem</i>	Specifies the URL of the CA on which your device should send certificate requests. <p>An IPv6 address can be added in the URL enclosed in brackets. For example: http:// [2001:DB8:1:1::1]:80.</p>

	Command or Action	Purpose
		The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.
Step 5	rsa keypair <i>label</i>	Specifies which key pair to associate with the certificate.
Step 6	serial-number none	The none keyword specifies that a serial number will not be included in the certificate request.
Step 7	ip-address none	The none keyword specifies that no IP address should be included in the certificate request.
Step 8	revocation-check <i>crl</i>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 9	exit	Exits Global Configuration mode.
Step 10	crypto pki authenticate <i>name</i>	Retrieves the CA certificate and authenticates it.
Step 11	crypto pki enroll <i>name</i>	<p>Generates certificate request and displays the request for copying and pasting into the certificate server.</p> <p>Enter enrollment information when you are prompted. For example, specify whether to include the device FQDN and IP address in the certificate request.</p> <p>You are also given the choice about displaying the certificate request to the console terminal.</p> <p>The base-64 encoded certificate with or without PEM headers as requested is displayed.</p>
Step 12	crypto pki import <i>name certificate</i>	<p>Imports a certificate via TFTP at the console terminal, which retrieves the granted certificate.</p> <p>The device attempts to retrieve the granted certificate via TFTP using the same filename used to send the request, except the extension is changed from “.req” to “.crt”. For usage key certificates, the extensions “-sign.crt” and “-encr.crt” are used.</p> <p>The device parses the received files, verifies the certificates, and inserts the certificates into the internal certificate database on the switch.</p> <p>Note Some CAs ignore the usage key information in the certificate request and issue general purpose usage certificates. If your CA ignores the usage key information in the certificate request, only import the general purpose certificate. The router will not use one of the two key pairs generated.</p>
Step 13	exit	Exits Global Configuration mode.

	Command or Action	Purpose
Step 14	<code>show crypto pki certificate <i>trustpoint name</i></code>	Displays information about the certificate for the trust point.
Step 15	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring EAP-TLS Profile and 802.1x Credentials

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>eap profile <i>profile-name</i></code>	Configures EAP profile and enters EAP profile configuration mode.
Step 4	<code>method tls</code>	Enables EAP-TLS method on the device.
Step 5	<code>pki-trustpoint <i>name</i></code>	Sets the default PKI trustpoint.
Step 6	<code>exit</code>	Returns to global configuration mode.
Step 7	<code>dot1x credentials <i>profile-name</i></code>	Configures 802.1x credentials profile and enters dot1x credentials configuration mode.
Step 8	<code>username <i>username</i></code>	Sets the authentication user ID.
Step 9	<code>pki-trustpoint <i>name</i></code>	Sets the default PKI trustpoint.
Step 10	<code>end</code>	Returns to privileged EXEC mode.

Applying the 802.1x MKA MACsec Configuration on Interfaces

To apply MKA MACsec using EAP-TLS to interfaces, perform the following task:

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>interface <i>interface-id</i></code>	Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.

The **show macsec status interface *interface-id*** displays MACsec status information for the given interface.

```
Device# show macsec status interface te0/1/2
```

```
Capabilities:
Ciphers Supported:      GCM-AES-128 GCM-AES-256
Cipher:                GCM-AES-128
Confidentiality Offset: 0
Replay Window:        64
Delay Protect Enable:  FALSE
Access Control:       must-secure

Transmit SC:
  SCI:                74A2E6254C220012
  Transmitting:      TRUE
Transmit SA:
  Next PN:           412
  Delay Protect AN/nextPN: 99/0

Receive SC:
  SCI:                74A2E62544130013
  Receiving:         TRUE
Receive SA:
  Next PN:           64
  AN:                0
  Delay Protect AN/LPN: 0/0
```

The **show access-session interface *interface-id* details** displays detailed information about the access session for the given interface.

```
Device# show access-session interface tel/0/1 details
```

```
Interface: TenGigabitEthernet1/0/1
  IIF-ID: 0x17298FCD
  MAC Address: f8a5.c592.13e4
  IPv6 Address: Unknown
  IPv4 Address: Unknown
  User-Name: DOT1XCRED
  Status: Authorized
  Domain: DATA
  Oper host mode: multi-host
  Oper control dir: both
  Session timeout: N/A
  Common Session ID: 00000000000000BB72E8AFA
  Acct Session ID: Unknown
  Handle: 0xc3000001
  Current Policy: MUSTS_1

Local Policies:
  Security Policy: Must Secure
  Security Status: Link Secured

Server Policies:

Method status list:
  Method      State
  dotlxCred   Authc Success
  dotlxCred   Authc Success
```

Configuring Cisco TrustSec MACsec

Configuring Cisco TrustSec Credentials on the Switch

To enable Cisco TrustSec features, you must create Cisco TrustSec credentials on the switch to use in other TrustSec configurations. Beginning in privilege EXEC mode, follow these steps to configure Cisco TrustSec credentials.

SUMMARY STEPS

1. `cts credentials id device-id password cts-password`
2. `show cts credentials`
3. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>cts credentials id <i>device-id</i> password <i>cts-password</i></code></p> <p>Example:</p> <pre>Switch# cts credentials id trustsec password mypassword</pre>	<p>Specifies the Cisco TrustSec credentials for this switch to use when authenticating with other Cisco TrustSec devices with EAP-FAST.</p> <ul style="list-style-type: none"> • id <i>device-id</i>—Specifies a Cisco TrustSec device ID for the switch. The device-id argument has a maximum length of 32 characters and is case sensitive • password <i>cts-password</i>—Specifies the Cisco TrustSec password for the device.
Step 2	<p><code>show cts credentials</code></p> <p>Example:</p> <pre>Switch# show cts credentials</pre>	(Optional) Displays Cisco TrustSec credentials configured on the switch.
Step 3	<p><code>copy running-config startup-config</code></p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Example

To delete the Cisco TrustSec credentials, enter the **clear cts credentials** privileged EXEC command.

This example shows how to create Cisco TrustSec credentials.

```
Switch# cts credentials id trustsec password mypassword
CTS device ID and password have been inserted in the local keystore. Please make
sure that the same ID and password are configured in the server database.
```

```
Switch# show cts credentials
CTS password is defined in keystore, device-id = trustsec
```

What to do next

Before you configure Cisco TrustSec MACsec authentication, you should configure Cisco TrustSec seed and non-seed devices. For 802.1x mode, you must configure at least one seed device, that device closest to the access control system (ACS). See this section in the Cisco TrustSec Configuration

Guide:http://www.cisco.com/en/US/docs/switches/lan/trustsec/configuration/guide/ident-conn_config.html

Configuring Cisco TrustSec Switch-to-Switch Link Security in 802.1x Mode

Before you begin

You enable Cisco TrustSec link layer switch-to-switch security on an interface that connects to another Cisco TrustSec device. When configuring Cisco TrustSec in 802.1x mode on an interface, follow these guidelines:

- To use 802.1x mode, you must globally enable 802.1x on each device. For more information 802.1x, see the [Configuring IEEE 802.1x Port-Based Authentication](#) chapter.
- If you select GCM as the SAP operating mode, you must have a MACsec encryption software license from Cisco. MACsec is supported on Catalyst 3560cx universal IP base and IP services licenses. It is not supported with the NPE license or with a LAN base service image.

If you select GCM without the required license, the interface is forced to a link-down state.

Beginning in privilege EXEC mode, follow these steps to configure Cisco TrustSec switch-to-switch link layer security with 802.1x:

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **cts dot1x**
4. **sap mode-listmode1 [mode2 [mode3 [mode4]]]**
5. **no propagate sgt**
6. **exit**
7. **end**
8. **show cts interface [*interface-id* | brief |summary]**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i>	Note Enters interface configuration mode.

	Command or Action	Purpose
	Example: Switch(config)# interface tengigabitethernet 1/1/2	
Step 3	cts dot1x Example: Switch(config-if)# cts dot1x	Configures the interface to perform NDAC authentication.
Step 4	sap mode-listmode1 [mode2 [mode3 [mode4]]] Example: Switch(config-if-cts-dot1x)# sap mode-list gcm-encrypt null no-encap	(Optional) Configures the SAP operation mode on the interface. The interface negotiates with the peer for a mutually acceptable mode. Enter the acceptable modes in your order of preference. Choices for <i>mode</i> are: <ul style="list-style-type: none"> • gcm-encrypt—Authentication and encryption <p>Note Select this mode for MACsec authentication and encryption if your software license supports MACsec encryption.</p> <ul style="list-style-type: none"> • gmac—Authentication, no encryption • no-encap—No encapsulation • null—Encapsulation, no authentication or encryption <p>Note If the interface is not capable of data link encryption, no-encap is the default and the only available SAP operating mode. SGT is not supported.</p> <p>Note Although visible in the CLI help, the timer reauthentication and propagate sgt keywords are not supported.</p>
Step 5	no propagate sgt Example: Switch(config-if-cts-dot1x)# no propagate sgt	The switch (Catalyst 3560cx) does not support SGT tagging. This command disables propagation of SGT tag on the CTS link. It is mandatory that for the peer switch also to have "no propagate sgt" configured for the traffic to flow properly over the CTS link.
Step 6	exit Example: Switch(config-if-cts-dot1x)# exit	Exits Cisco TrustSec 802.1x interface configuration mode.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	show cts interface [<i>interface-id</i> brief summary]	(Optional) Verify the configuration by displaying TrustSec-related interface characteristics.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Example

This example shows how to enable Cisco TrustSec authentication in 802.1x mode on an interface using GCM as the preferred SAP mode:

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts dot1x
Switch(config-if-cts-dot1x)# sap mode-list gcm-encrypt null no-encap
Switch(config-if-cts-dot1x)# no propagate sgt
Switch(config-if-cts-dot1x)# exit
Switch(config-if)# end
```

Configuring Cisco TrustSec Switch-to-Switch Link Security in Manual Mode

Before you begin

When manually configuring Cisco TrustSec on an interface, consider these usage guidelines and restrictions:

- If no SAP parameters are defined, Cisco TrustSec encapsulation or encryption is not performed.
- If you select GCM as the SAP operating mode, you must have a MACsec Encryption software license from Cisco. If you select GCM without the required license, the interface is forced to a link-down state.
- These protection levels are supported when you configure SAP pairwise master key (sap pmk):
 - SAP is not configured—no protection.
 - **sap mode-list gcm-encrypt gmac no-encap**—protection desirable but not mandatory.
 - **sap mode-list gcm-encrypt gmac**—confidentiality preferred and integrity required. The protection is selected by the supplicant according to supplicant preference.
 - **sap mode-list gmac**—integrity only.
 - **sap mode-list gcm-encrypt**—confidentiality required.
 - **sap mode-list gmac gcm-encrypt**—integrity required and preferred, confidentiality optional.

Beginning in privileged EXEC mode, follow these steps to manually configure Cisco TrustSec on an interface to another Cisco TrustSec device:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **cts manual**
4. **sap pmk** *key* [**mode-list** *mode1* [*mode2* [*mode3* [*mode4*]]]]
5. **no propagate sgt**
6. **exit**
7. **end**
8. **show cts interface** [*interface-id* | **brief** | **summary**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface tengigabitethernet 1/1/2	Note Enters interface configuration mode.
Step 3	cts manual Example: Switch(config-if)# cts manual	Enters Cisco TrustSec manual configuration mode.
Step 4	sap pmk <i>key</i> [mode-list <i>mode1</i> [<i>mode2</i> [<i>mode3</i> [<i>mode4</i>]]]] Example: Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap	(Optional) Configures the SAP pairwise master key (PMK) and operation mode. SAP is disabled by default in Cisco TrustSec manual mode. <ul style="list-style-type: none"> • <i>key</i>—A hexadecimal value with an even number of characters and a maximum length of 32 characters. The SAP operation mode options: <ul style="list-style-type: none"> • gcm-encrypt—Authentication and encryption <p>Note Select this mode for MACsec authentication and encryption if your software license supports MACsec encryption.</p> • gmac—Authentication, no encryption • no-encap—No encapsulation • null—Encapsulation, no authentication or encryption

	Command or Action	Purpose
		<p>Note If the interface is not capable of data link encryption, no-encap is the default and the only available SAP operating mode. SGT is not supported.</p>
Step 5	<p>no propagate sgt</p> <p>Example:</p> <pre>Switch(config-if-cts-manual)# no propagate sgt</pre>	Use the no form of this command when the peer is incapable of processing a SGT. The no propagate sgt command prevents the interface from transmitting the SGT to the peer.
Step 6	<p>exit</p> <p>Example:</p> <pre>Switch(config-if-cts-manual)# exit</pre>	Exits Cisco TrustSec 802.1x interface configuration mode.
Step 7	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show cts interface [<i>interface-id</i> brief summary]</p>	(Optional) Verify the configuration by displaying TrustSec-related interface characteristics.

Example

This example shows how to configure Cisco TrustSec authentication in manual mode on an interface:

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# end
```

Configuration Examples for Configuring MACsec Encryption

Example: Configuring MACsec on an Interface

Configuring MACsec on an Interface

```
Switch(config)# interface GigabitEthernet1/0/25
Switch(config-if)# switchport access vlan 10
Switch(config-if)# switchport mode access
Switch(config-if)# macsec
Switch(config-if)# authentication event linksec fail action authorize vlan 2
Switch(config-if)# authentication host-mode multi-domain
Switch(config-if)# authentication linksec policy must-secure
```

```

Switch(config-if)# authentication port-control auto
Switch(config-if)# authentication periodic
Switch(config-if)# authentication timer reauthenticate
Switch(config-if)# authentication violation protect
Switch(config-if)# mka policy replay-policy
Switch(config-if)# dot1x pae authenticator
Switch(config-if)# spanning-tree portfast
Switch(config-if)# end
Switch# show authentication session interface gigabitethernet1/0/5

```

```

Interface MAC Address Method Domain Status Fg Session ID
-----
Gi1/0/5 88f0.7788.9205 dot1x VOICE Auth 1E0000010000001300030B0F
Gi1/0/5 000c.2923.6ff1 dot1x DATA Auth 1E0000010000001400030D80

```

Key to Session Events Blocked Status Flags:

```

A - Applying Policy (multi-line status for details)
D - Awaiting Deletion
F - Final Removal in progress
I - Awaiting IIF ID allocation
N - Waiting for AAA to come up
P - Pushed Session
R - Removing User Profile (multi-line status for details)
U - Applying User Profile (multi-line status for details)
X - Unknown Blocker

```

Runnable methods list:

```

Handle Priority Name
7 5 dot1x
21 10 mab

```

```

19 15 webauth

```

```

Switch# show authentication session interface gigabitethernet1/0/5 details

```

```

Interface: GigabitEthernet1/0/5
MAC Address: 88f0.7788.9205
IPv6 Address: Unknown
IPv4 Address: Unknown
User-Name: CP-9971-SEP88F077889205
Status: Authorized
Domain: VOICE
Oper host mode: multi-domain
Oper control dir: both
Session timeout: N/A
Common Session ID: 1E0000010000001300030B0F
Acct Session ID: Unknown
Handle: 0xC0000006
Current Policy: POLICY_Gi1/0/5

```

Local Policies:

```

Service Template: DEFAULT_LINKSEC_POLICY_SHOULD_SECURE (priority 150)
Security Policy: Should Secure
Security Status: Link Unsecure

```

Server Policies:

Method status list:

Method State

```

dot1x Authc Success

-----
Interface: GigabitEthernet1/0/5
MAC Address: 000c.2923.6ff1
IPv6 Address: Unknown
IPv4 Address: 172.30.30.50
User-Name: dataMustSecure
Status: Authorized
Domain: DATA
Oper host mode: multi-domain
Oper control dir: both
Session timeout: N/A
Common Session ID: 1E0000010000001400030D80
Acct Session ID: Unknown
Handle: 0x22000007
Current Policy: POLICY_Gil/0/5

Local Policies:
Service Template: DEFAULT_LINKSEC_POLICY_SHOULD_SECURE (priority 150)
Security Policy: Should Secure

Security Status: Link Secured

Server Policies:

Method status list:
Method State

dot1x Authc Success

Switch#
Switch# show macsec interface gigabitethernet1/0/5
MACsec is enabled
Replay protect : enabled
Replay window : 0
Include SCI : yes
Use ES Enable : no
Use SCB Enable : no
Admin Pt2Pt MAC : forceTrue(1)
Pt2Pt MAC Operational : no
Cipher : GCM-AES-128
Confidentiality Offset : 0

Capabilities
Identifier :
Name :
ICV length : 16
Data length change supported: yes
Max. Rx SA : 8
Max. Tx SA : 8
Max. Rx SC : 4
Max. Tx SC : 4
Validate Frames : strict
PN threshold notification support : Yes
Ciphers supported : GCM-AES-128

Transmit Secure Channels
SCI : 547C69B687850002
SC state : inUse(1)
Elapsed time : 16:36:44
Start time : 7w0d
Current AN: 0

```

```
Previous AN: -
Next PN: 0
SA State: inUse(1)
Confidentiality : no
SAK Unchanged : no
SA Create time : 00:09:21
SA Start time : 7w0d
SC Statistics
Auth-only Pkts : 0
Auth-only Bytes : 0
Encrypt Pkts : 52960
Encrypt Bytes : 0
SA Statistics
Auth-only Pkts : 0
Encrypt Pkts : 52960

Port Statistics

Receive Secure Channels
SCI : 000C29236FF10000
SC state : inUse(1)
Elapsed time : 16:36:44
Start time : 7w0d
Current AN: 0
Previous AN: -
Next PN: 0
RX SA Count: 0
SA State: inUse(1)
SAK Unchanged : no
SA Create time : 00:09:19
SA Start time : 7w0d
SC Statistics
Notvalid pkts 0
Invalid pkts 0
Valid pkts 9914
Valid bytes 0
Late pkts 0
Uncheck pkts 0
Delay pkts 0
UnusedSA pkts 0
NousingSA pkts 0
Decrypt bytes 0
SA Statistics
Notvalid pkts 0
Invalid pkts 0
Valid pkts 9914
UnusedSA pkts 0
NousingSA pkts 0

Port Statistics

Switch#
```

Configuration Examples for MACsec MKA using EAP-TLS

Example: Enrolling the Certificate

```
Configure Crypto PKI Trustpoint:
```

Example: Enabling 802.1x Authentication and AAA Configuration

```
crypto pki trustpoint POLESTAR-IOS-CA
  enrollment terminal
  subject-name CN=ASR1000x1@polestar.com, C=IN, ST=KA, OU=ENG,O=Polestar
  revocation-check none
  rsa-keypair mkaioscarsa
  storage nvram:
!
```

Manual Installation of Root CA certificate:

```
crypto pki authenticate POLESTAR-IOS-CA
```

Example: Enabling 802.1x Authentication and AAA Configuration

```
aaa new-model
dot1x system-auth-control
radius server ISE
  address ipv4 <ISE ipv4 address> auth-port 1645 acct-port 1646
  automate-tester username dummy
  key dummy123
  radius-server deadtime 2
!
aaa group server radius ISEGRP
  server name ISE
!
aaa authentication dot1x default group ISEGRP
aaa authorization network default group ISEGRP
```

Example: Configuring EAP-TLS Profile and 802.1X Credentials

```
eap profile EAPTLS-PROF-IOSCA
  method tls
  pki-trustpoint POLESTAR-IOS-CA
!
dot1x credentials EAPTLSCRED-IOSCA
  username asr1000@polestar.company.com
  pki-trustpoint POLESTAR-IOS-CA
!
```

Example: Applying 802.1X, PKI, and MACsec Configuration on the Interface

```
interface TenGigabitEthernet0/1
  macsec network-link
  authentication periodic
  authentication timer reauthenticate <reauthentication interval>
  access-session host-mode multi-host
  access-session closed
  access-session port-control auto
  dot1x pae both
  dot1x credentials EAPTLSCRED-IOSCA
  dot1x supplicant eap profile EAPTLS-PROF-IOSCA
  service-policy type control subscriber DOT1X_POLICY_RADIUS
```


Cisco TrustSec Switch-to-Switch Link Security Configuration Example

This example shows the configuration necessary for a seed and non-seed device for Cisco TrustSec switch-to-switch security. You must configure the AAA and RADIUS for link security. In this example, ACS-1 through ACS-3 can be any server names and cts-radius is the Cisco TrustSec server.

Seed Device Configuration:

```
Switch(config)# aaa new-model
Switch(config)# radius server ACS-1
Switch(config-radius-server)# address ipv4 10.5.120.12 auth-port 1812 acct-port 1813
Switch(config-radius-server)# pac key cisco123
Switch(config-radius-server)# exit
Switch(config)# radius server ACS-2
Switch(config-radius-server)# address ipv4 10.5.120.14 auth-port 1812 acct-port 1813
Switch(config-radius-server)# pac key cisco123
Switch(config-radius-server)# exit
Switch(config)# radius server ACS-3
Switch(config-radius-server)# address ipv4 10.5.120.15 auth-port 1812 acct-port 1813
Switch(config-radius-server)# pac key cisco123
Switch(config-radius-server)# exit
Switch(config)# aaa group server radius cts-radius
Switch(config-sg-radius)# server name ACS-1
Switch(config-sg-radius)# server name ACS-2
Switch(config-sg-radius)# server name ACS-3
Switch(config-sg-radius)# exit
Switch(config)# aaa authentication login default none
Switch(config)# aaa authentication dot1x default group cts-radius
Switch(config)# aaa authorization network cts-radius group cts-radius
Switch(config)# aaa session-id common
Switch(config)# cts authorization list cts-radius
Switch(config)# dot1x system-auth-control

Switch(config)# interface gil1/1/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# cts dot1x
Switch(config-if-cts-dot1x)# sap mode-list gcm-encrypt gmac
Switch(config-if-cts-dot1x)# no propagate sgt
Switch(config-if-cts-dot1x)# exit
Switch(config-if)# exit

Switch(config)# interface gil1/1/4
Switch(config-if)# switchport mode trunk
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt gmac
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# exit
```

```
Switch(config)# radius-server vsa send authentication
Switch(config)# exit
Switch# cts credentials id cts-36 password trustsec123
```

Non-Seed Device:

```
Switch(config)# aaa new-model
Switch(config)# aaa session-id common
Switch(config)# dot1x system-auth-control

Switch(config)# interface gil/1/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# shutdown
Switch(config-if)# cts dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit

Switch(config)# interface gil/1/4
Switch(config-if)# switchport mode trunk
Switch(config-if)# shutdown
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
gmac
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# exit

Switch(config)# radius-server vsa send authentication
Switch(config)# end
Switch# cts credentials id cts-72 password trustsec123
```



CHAPTER 70

Web-Based Authentication

This chapter describes how to configure web-based authentication on the device. It contains these sections:

- [Finding Feature Information, on page 1451](#)
- [Web-Based Authentication Overview, on page 1451](#)
- [How to Configure Web-Based Authentication, on page 1460](#)
- [Verifying Web-Based Authentication Status, on page 1474](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Web-Based Authentication Overview

Use the web-based authentication feature, known as web authentication proxy, to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.

When you initiate an HTTP session, web-based authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the web-based authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

If authentication succeeds, web-based authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server.

If authentication fails, web-based authentication forwards a Login-Fail HTML page to the user, prompting the user to retry the login. If the user exceeds the maximum number of attempts, web-based authentication forwards a Login-Expired HTML page to the host, and the user is placed on a watch list for a waiting period.



Note HTTPS traffic interception for central web authentication redirect is not supported.



Note You should use global parameter-map (for method-type, custom, and redirect) only for using the same web authentication methods like consent, web consent, and webauth, for all the clients and SSIDs. This ensures that all the clients have the same web-authentication method.

If the requirement is to use Consent for one SSID and Web-authentication for another SSID, then you should use two named parameter-maps. You should configure Consent in first parameter-map and configure webauth in second parameter-map.



Note The traceback that you receive when webauth client tries to do authentication does not have any performance or behavioral impact. It happens rarely when the context for which FFM replied back to EPM for ACL application is already dequeued (possibly due to timer expiry) and the session becomes 'unauthorized'.

Based on where the web pages are hosted, the local web authentication can be categorized as follows:

- *Internal*—The internal default HTML pages (Login, Success, Fail, and Expire) in the controller are used during the local web authentication.
- *Customized*—The customized web pages (Login, Success, Fail, and Expire) are downloaded onto the controller and used during the local web authentication.
- *External*—The customized web pages are hosted on the external web server instead of using the in-built or custom web pages.

Based on the various web authentication pages, the types of web authentication are as follows:

- *Webauth*—This is a basic web authentication. Herein, the controller presents a policy page with the user name and password. You need to enter the correct credentials to access the network.
- *Consent or web-passthrough*—Herein, the controller presents a policy page with the Accept or Deny buttons. You need to click the Accept button to access the network.
- *Webconsent*—This is a combination of webauth and consent web authentication types. Herein, the controller presents a policy page with Accept or Deny buttons along with user name or password. You need to enter the correct credentials and click the Accept button to access the network.

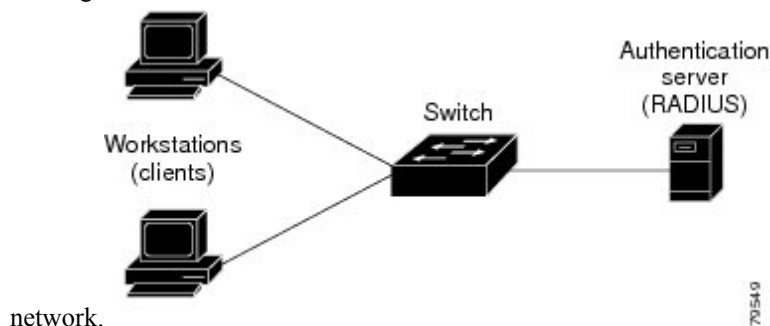
Device Roles

With web-based authentication, the devices in the network have these specific roles:

- *Client*—The device (workstation) that requests access to the LAN and the services and responds to requests from the switch. The workstation must be running an HTML browser with Java Script enabled.
- *Authentication server*—Authenticates the client. The authentication server validates the identity of the client and notifies the switch that the client is authorized to access the LAN and the switch services or that the client is denied.
- *Switch*—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client.

Figure 114: Web-Based Authentication Device Roles

This figure shows the roles of these devices in a



Host Detection

The switch maintains an IP device tracking table to store information about detected hosts.



Note By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.

For Layer 2 interfaces, web-based authentication detects IP hosts by using these mechanisms:

- ARP based trigger—ARP redirect ACL allows web-based authentication to detect hosts with a static IP address or a dynamic IP address.
- Dynamic ARP inspection
- DHCP snooping—Web-based authentication is notified when the switch creates a DHCP-binding entry for the host.

Session Creation

When web-based authentication detects a new host, it creates a session as follows:

- Reviews the exception list.
If the host IP is included in the exception list, the policy from the exception list entry is applied, and the session is established.
- Reviews for authorization bypass
If the host IP is not on the exception list, web-based authentication sends a nonresponsive-host (NRH) request to the server.
If the server response is access accepted, authorization is bypassed for this host. The session is established.
- Sets up the HTTP intercept ACL
If the server response to the NRH request is access rejected, the HTTP intercept ACL is activated, and the session waits for HTTP traffic from the host.

Authentication Process

When you enable web-based authentication, these events occur:

- The user initiates an HTTP session.
- The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
- If the authentication succeeds, the switch downloads and activates the user's access policy from the authentication server. The login success page is sent to the user.
- If the authentication fails, the switch sends the login fail page. The user retries the login. If the maximum number of attempts fails, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.
- If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user.
- The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface, or when the host does not send any traffic within the idle timeout on a Layer 3 interface.
- The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface.
- The feature applies the downloaded timeout or the locally configured session timeout.
- If the terminate action is RADIUS, the feature sends a nonresponsive host (NRH) request to the server. The terminate action is included in the response from the server.
- If the terminate action is default, the session is dismantled, and the applied policy is removed.

Local Web Authentication Banner

With Web Authentication, you can create a default and customized web-browser banners that appears when you log in to a switch.

The banner appears on both the login page and the authentication-result pop-up pages. The default banner messages are as follows:

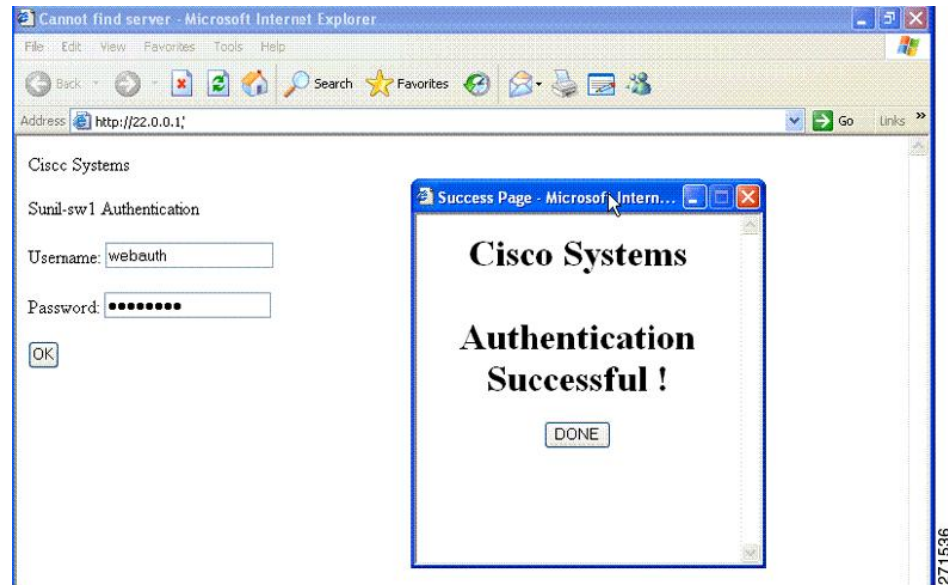
- *Authentication Successful*
- *Authentication Failed*
- *Authentication Expired*

The Local Web Authentication Banner can be configured in legacy and new-style (Session-aware) CLIs as follows:

- Legacy mode—Use the **ip admission auth-proxy-banner http** global configuration command.
- New-style mode—Use the **parameter-map type webauth global banner** global configuration command.

The default banner *Cisco Systems* and *Switch host-name Authentication* appear on the Login Page. *Cisco Systems* appears on the authentication result pop-up page.

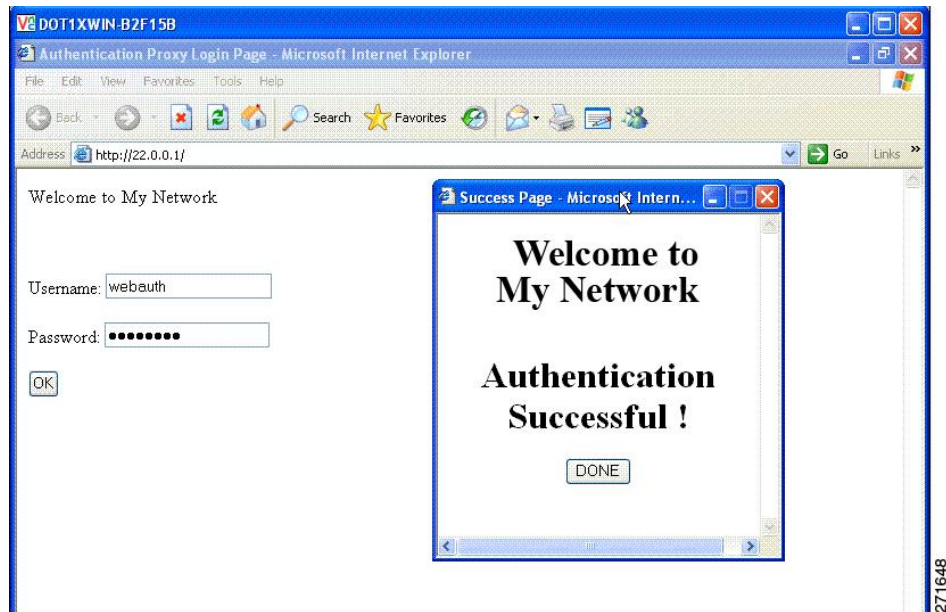
Figure 115: Authentication Successful Banner



The banner can be customized as follows:

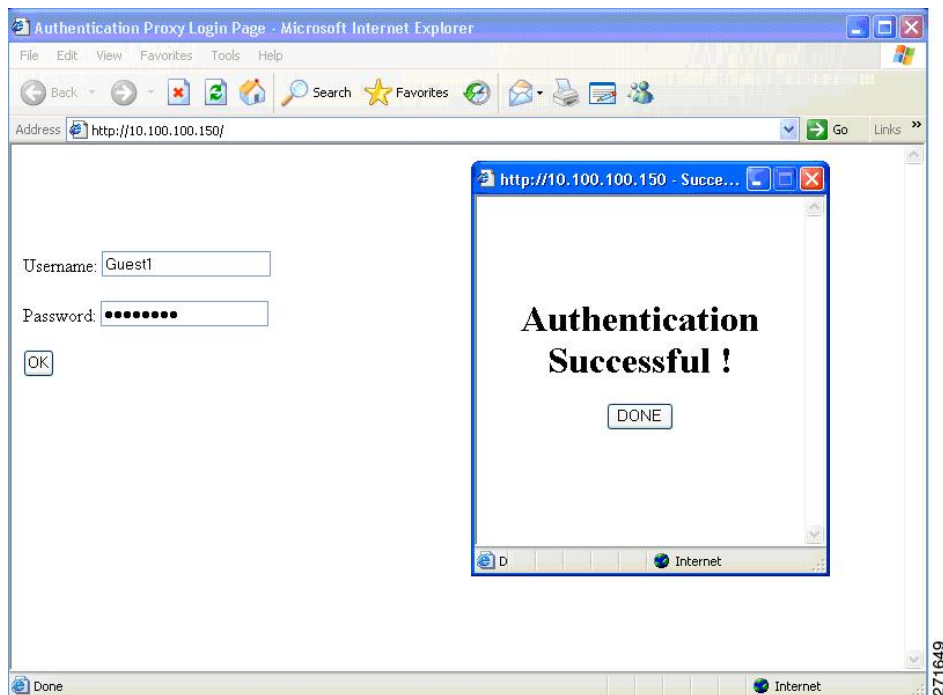
- Add a message, such as switch, router, or company name to the banner:
 - Legacy mode—Use the **ip admission auth-proxy-banner http banner-text** global configuration command.
 - New-style mode—Use the **parameter-map type webauth global banner** global configuration command.
- Add a logo or text file to the banner:
 - Legacy mode—Use the **ip admission auth-proxy-banner http file-path** global configuration command.
 - New-style mode—Use the **parameter-map type webauth global banner** global configuration command.

Figure 116: Customized Web Banner



If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login screen, and no banner appears when you log into the switch.

Figure 117: Login Screen With No Banner



Web Authentication Customizable Web Pages

During the web-based authentication process, the switch internal HTTP server hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify you of these four-authentication process states:

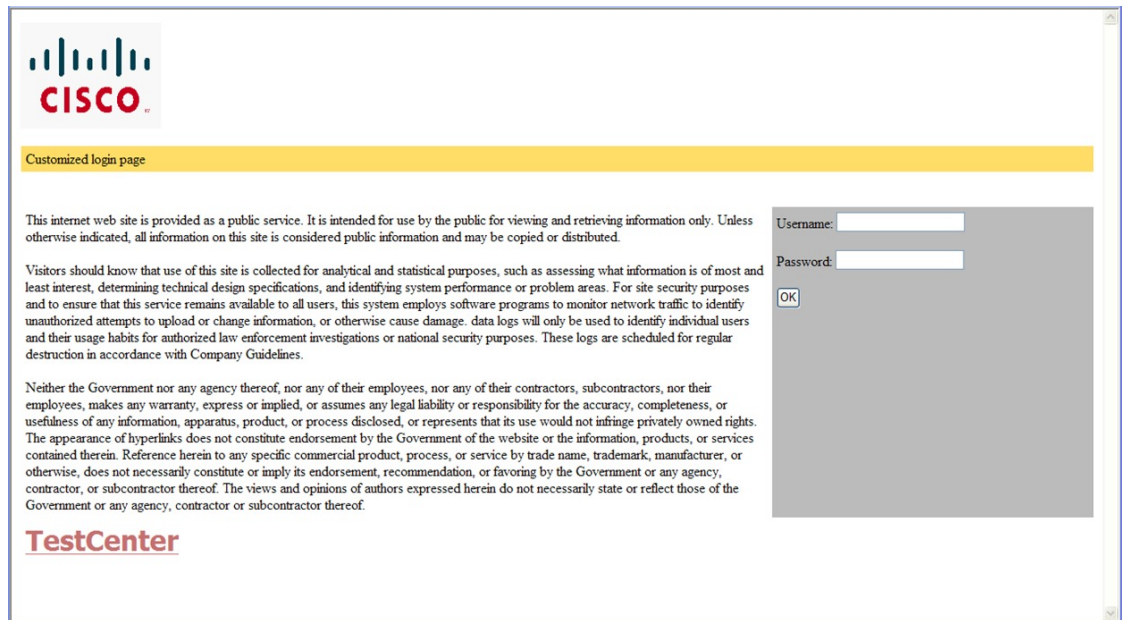
- Login—Your credentials are requested.
- Success—The login was successful.
- Fail—The login failed.
- Expire—The login session has expired because of excessive login failures.

Guidelines

- You can substitute your own HTML pages for the default internal HTML pages.
- You can use a logo or specify text in the *login*, *success*, *failure*, and *expire* web pages.
- On the banner page, you can specify text in the login page.
- The pages are in HTML.
- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, <http://www.cisco.com>). An incomplete URL might cause *page not found* or similar errors on a web browser.
- If you configure web pages for HTTP authentication, they must include the appropriate HTML commands (for example, to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice).
- The CLI command to redirect users to a specific URL is not available when the configured login form is enabled. The administrator should ensure that the redirection is configured in the web page.
- If the CLI command redirecting users to specific URL after authentication occurs is entered and then the command configuring web pages is entered, the CLI command redirecting users to a specific URL does not take effect.
- Configured web pages can be copied to the switch boot flash or flash.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the active switch or a member switch).
- You must configure all four pages.
- The banner page has no effect if it is configured with the web page.
- All of the logo files (image, flash, audio, video, and so on) that are stored in the system directory (for example, flash, disk0, or disk) and that must be displayed on the login page must use *web_auth_<filename>* as the file name.
- The configured authentication proxy feature supports both HTTP and SSL.

You can substitute your HTML pages for the default internal HTML pages. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

Figure 118: Customizable Authentication Page



Authentication Proxy Web Page Guidelines

When configuring customized authentication proxy web pages, follow these guidelines:

- To enable the custom web pages feature, specify all four custom HTML files. If you specify fewer than four files, the internal default HTML pages are used.
- The four custom HTML files must be present on the flash memory of the switch. The maximum size of each HTML file is 8 KB.
- Any images on the custom pages must be on an accessible HTTP server. Configure an intercept ACL within the admission rule.
- Any external link from a custom page requires configuration of an intercept ACL within the admission rule.
- To access a valid DNS server, any name resolution required for external links or images requires configuration of an intercept ACL within the admission rule.
- If the custom web pages feature is enabled, a configured auth-proxy-banner is not used.
- If the custom web pages feature is enabled, the redirection URL for successful login feature is not available.
- To remove the specification of a custom file, use the **no** form of the command.

Because the custom login page is a public web form, consider these guidelines for the page:

- The login form must accept user entries for the username and password and must show them as **uname** and **pwd**.
- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.

Redirection URL for Successful Login Guidelines

When configuring a redirection URL for successful login, consider these guidelines:

- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.
- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used.
- To remove the specification of a redirection URL, use the **no** form of the command.
- If the redirection URL is required after the web-based authentication client is successfully authenticated, then the URL string must start with a valid URL (for example, http://) followed by the URL information. If only the URL is given without http://, then the redirection URL on successful authentication might cause page not found or similar errors on a web browser.

Web-based Authentication Interactions with Other Features

Port Security

You can configure web-based authentication and port security on the same port. Web-based authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through the port.

LAN Port IP

You can configure LAN port IP (LPIP) and Layer 2 web-based authentication on the same port. The host is authenticated by using web-based authentication first, followed by LPIP posture validation. The LPIP host policy overrides the web-based authentication host policy.

If the web-based authentication idle timer expires, the NAC policy is removed. The host is authenticated, and posture is validated again.

Gateway IP

You cannot configure Gateway IP (GWIP) on a Layer 3 VLAN interface if web-based authentication is configured on any of the switch ports in the VLAN.

You can configure web-based authentication on the same Layer 3 interface as Gateway IP. The host policies for both features are applied in software. The GWIP policy overrides the web-based authentication host policy.

ACLs

If you configure a VLAN ACL or a Cisco IOS ACL on an interface, the ACL is applied to the host traffic only after the web-based authentication host policy is applied.

For Layer 2 web-based authentication, it is more secure, though not required, to configure a port ACL (PACL) as the default access policy for ingress traffic from hosts connected to the port. After authentication, the web-based authentication host policy overrides the PACL. The Policy ACL is applied to the session even if there is no ACL configured on the port.

You cannot configure a MAC ACL and web-based authentication on the same interface.

You cannot configure web-based authentication on a port whose access VLAN is configured for VACL capture.

Context-Based Access Control

Web-based authentication cannot be configured on a Layer 2 port if context-based access control (CBAC) is configured on the Layer 3 VLAN interface of the port VLAN.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

How to Configure Web-Based Authentication

Default Web-Based Authentication Configuration

The following table shows the default web-based authentication configuration.

Table 147: Default Web-based Authentication Configuration

Feature	Default Setting
AAA	Disabled
RADIUS server <ul style="list-style-type: none"> • IP address • UDP authentication port • Key 	<ul style="list-style-type: none"> • None specified • 1645 • None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Web-Based Authentication Configuration Guidelines and Restrictions

- Web-based authentication is an ingress-only feature.
- You can configure web-based authentication only on access ports. Web-based authentication is not supported on trunk ports, EtherChannel member ports, or dynamic trunk ports.
- External web authentication, where the switch redirects a client to a particular host or web server for displaying login message, is not supported.
- You cannot authenticate hosts on Layer 2 interfaces with static ARP cache assignment. These hosts are not detected by the web-based authentication feature because they do not send ARP messages.
- By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.
- You must enable SISF-Based device tracking to use web-based authentication. By default, SISF-Based device tracking is disabled on a switch.

- You must configure at least one IP address to run the switch HTTP server. You must also configure routes to reach each host IP address. The HTTP server sends the HTTP login page to the host.
- Hosts that are more than one hop away might experience traffic disruption if an STP topology change results in the host traffic arriving on a different port. This occurs because the ARP and DHCP updates might not be sent after a Layer 2 (STP) topology change.
- Web-based authentication does not support VLAN assignment as a downloadable-host policy.
- Web-based authentication supports IPv6 in Session-aware policy mode. IPv6 Web-authentication requires at least one IPv6 address configured on the switch and IPv6 Snooping configured on the switchport.
- Web-based authentication and Network Edge Access Topology (NEAT) are mutually exclusive. You cannot use web-based authentication when NEAT is enabled on an interface, and you cannot use NEAT when web-based authentication is running on an interface.
- Identify the following RADIUS security server settings that will be used while configuring switch-to-RADIUS-server communication:
 - Host name
 - Host IP address
 - Host name and specific UDP port numbers
 - IP address and specific UDP port numbers

The combination of the IP address and UDP port number creates a unique identifier, that enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication) the second host entry that is configured functions as the failover backup to the first one. The RADIUS host entries are chosen in the order that they were configured.

- When you configure the RADIUS server parameters:
 - Specify the **key string** on a separate command line.
 - For **key string**, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server.
 - When you specify the **key string**, use spaces within and at the end of the key. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.
 - You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using with the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server transmit**, and the **radius-server key** global configuration commands.



Note You need to configure some settings on the RADIUS server, including: the switch IP address, the key string to be shared by both the server and the switch, and the downloadable ACL (DAACL). For more information, see the RADIUS server documentation.

Configuring the Authentication Rule and Interfaces

Follow these steps to configure the authentication rule and interfaces:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission name *name* proxy http**
4. **interface *type slot/port***
5. **ip access-group *name***
6. **ip admission name**
7. **exit**
8. **ip device tracking**
9. **end**
10. **show ip admission status**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip admission name <i>name</i> proxy http Example: <pre>Switch(config)# ip admission name webauth1 proxy http</pre>	Configures an authentication rule for web-based authorization.
Step 4	interface <i>type slot/port</i> Example: <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode and specifies the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication. <i>type</i> can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 5	ip access-group <i>name</i> Example:	Applies the default ACL.

	Command or Action	Purpose
	<code>Switch(config-if)# ip access-group webauthag</code>	
Step 6	ip admission name Example: <code>Switch(config)# ip admission name</code>	Configures an authentication rule for web-based authorization for the interface.
Step 7	exit Example: <code>Switch(config-if)# exit</code>	Returns to configuration mode.
Step 8	ip device tracking Example: <code>Switch(config)# ip device tracking</code>	Enables the IP device tracking table.
Step 9	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.
Step 10	show ip admission status Example: <code>Switch# show ip admission</code>	Displays the configuration.
Step 11	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring AAA Authentication

SUMMARY STEPS

1. `aaa new-model`
2. `aaa authentication login default group {tacacs+ | radius}`
3. `aaa authorization auth-proxy default group {tacacs+ | radius}`

DETAILED STEPS

	Command or Action	Purpose
Step 1	aaa new-model Example: <pre>Switch(config)# aaa new-model</pre>	Enables AAA functionality.
Step 2	aaa authentication login default group {tacacs+ radius} Example: <pre>Switch(config)# aaa authentication login default group tacacs+</pre>	Defines the list of authentication methods at login. named_authentication_list refers to any name that is not greater than 31 characters. AAA_group_name refers to the server group name. You need to define the server-group server_name at the beginning itself.
Step 3	aaa authorization auth-proxy default group {tacacs+ radius} Example: <pre>Switch(config)# aaa authorization auth-proxy default group tacacs+</pre>	Creates an authorization method list for web-based authorization.

Configuring Switch-to-RADIUS-Server Communication

Follow these steps to configure the RADIUS server parameters:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip radius source-interface vlan** *vlan interface number*
4. **radius-server host** *{hostname | ip-address}* **test username** *username*
5. **radius-server key** *string*
6. **radius-server dead-criteria tries** *num-tries*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip radius source-interface vlan <i>vlan interface number</i> Example: <pre>Switch(config)# ip radius source-interface vlan 80</pre>	Specifies that the RADIUS packets have the IP address of the indicated interface.
Step 4	radius-server host {<i>hostname</i> <i>ip-address</i>} test username <i>username</i> Example: <pre>Switch(config)# radius-server host 172.120.39.46 test username user1</pre>	<p>Specifies the host name or IP address of the remote RADIUS server.</p> <p>The test username <i>username</i> option enables automated testing of the RADIUS server connection. The specified <i>username</i> does not need to be a valid user name.</p> <p>The key option specifies an authentication and encryption key to use between the switch and the RADIUS server.</p> <p>To use multiple RADIUS servers, reenter this command for each server.</p>
Step 5	radius-server key <i>string</i> Example: <pre>Switch(config)# radius-server key rad123</pre>	Configures the authorization and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 6	radius-server dead-criteria tries <i>num-tries</i> Example: <pre>Switch(config)# radius-server dead-criteria tries 30</pre>	Specifies the number of unanswered sent messages to a RADIUS server before considering the server to be inactive. The range of <i>num-tries</i> is 1 to 100.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the HTTP Server

To use web-based authentication, you must enable the HTTP server within the Switch. You can enable the server for either HTTP or HTTPS.



Note The Apple psuedo-browser will not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Follow the procedure given below to enable the server for either HTTP or HTTPS:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http secure-server**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip http server Example: Switch(config)# ip http server	Enables the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 4	ip http secure-server Example: Switch(config)# ip http secure-server	Enables HTTPS. You can configure custom authentication proxy web pages or specify a redirection URL for successful login. Note To ensure secure authentication when you enter the ip http secure-server command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	<code>Switch(config)# end</code>	

Customizing the Authentication Proxy Web Pages

You can configure web authentication to display four substitute HTML pages to the user in place of the Switch default HTML pages during web-based authentication.

Follow these steps to specify the use of your custom authentication proxy web pages:

Before you begin

Store your custom HTML files on the Switch flash memory.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip admission proxy http login page file device:login-filename`
4. `ip admission proxy http success page file device:success-filename`
5. `ip admission proxy http failure page file device:fail-filename`
6. `ip admission proxy http login expired page file device:expired-filename`
7. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>ip admission proxy http login page file device:login-filename</code></p> <p>Example:</p> <pre>Switch(config)# ip admission proxy http login page file disk1:login.htm</pre>	<p>Specifies the location in the Switch memory file system of the custom HTML file to use in place of the default login page. The <i>device:</i> is flash memory.</p>

	Command or Action	Purpose
Step 4	ip admission proxy http success page file <i>device:success-filename</i> Example: <pre>Switch(config)# ip admission proxy http success page file disk1:success.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login success page.
Step 5	ip admission proxy http failure page file <i>device:fail-filename</i> Example: <pre>Switch(config)# ip admission proxy http fail page file disk1:fail.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login failure page.
Step 6	ip admission proxy http login expired page file <i>device:expired-filename</i> Example: <pre>Switch(config)# ip admission proxy http login expired page file disk1:expired.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login expired page.
Step 7	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Specifying a Redirection URL for Successful Login

Follow these steps to specify a URL to which the user is redirected after authentication, effectively replacing the internal Success HTML page:

SUMMARY STEPS

1. **configure terminal**
2. **configure terminal**
3. **ip admission proxy http success redirect *url-string***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	<code>Switch# configure terminal</code>	
Step 2	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 3	ip admission proxy http success redirect <i>url-string</i> Example: <code>Switch(config)# ip admission proxy http success redirect www.example.com</code>	Specifies a URL for redirection of the user in place of the default login success page.
Step 4	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.

Configuring Web-Based Authentication Parameters

Follow these steps to configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip admission max-login-attempts number`
4. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <code>Device> enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	ip admission max-login-attempts <i>number</i> Example: Device(config)# <code>ip admission max-login-attempts 10</code>	Sets the maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 4	exit Example: Device# <code>exit</code>	Exits global configuration mode and returns to privileged EXEC mode.

Configuring a Web-Based Authentication Local Banner

Follow these steps to configure a local banner on a switch that has web authentication configured.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip admission auth-proxy-banner http [banner-text | file-path]`
4. `end`
5. `show running-config`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip admission auth-proxy-banner http [<i>banner-text</i> <i>file-path</i>]	Enables the local banner.

	Command or Action	Purpose
	Example: <pre>Switch(config)# ip admission auth-proxy-banner http C My Switch C</pre>	(Optional) Create a custom banner by entering <i>C banner-text C</i> (where <i>C</i> is a delimiting character), or <i>file-path</i> that indicates a file (for example, a logo or text file) that appears in the banner.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Web-Based Authentication without SVI

You configure the web-based authentication without SVI feature to redirect the HTML login page to the client without creating an IP address in the routing table. These steps are optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type webauth global**
4. **l2-webauth-enabled**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	parameter-map type webauth global Example: Switch (config)# <code>parameter-map type webauth global</code>	Creates a parameter map and enters parameter-map webauth configuration mode. The specific configuration commands supported for a global parameter map defined with the global keyword differ from the commands supported for a named parameter map defined with the parameter-map-name argument.
Step 4	l2-webauth-enabled Example: Switch (config-params-parameter-map)# <code>l2-webauth-enabled</code>	Enables the web-based authentication without SVI feature
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Web-Based Authentication with VRF Aware

You configure the web-based authentication with VRF aware to redirect the HTML login page to the client. These steps are optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `parameter-map type webauth global`
4. `webauth-vrf-aware`
5. `end`

6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	parameter-map type webauth global Example: <pre>Switch (config)# parameter-map type webauth global</pre>	Creates a parameter map and enters parameter-map webauth configuration mode. The specific configuration commands supported for a global parameter map defined with the global keyword differ from the commands supported for a named parameter map defined with the parameter-map-name argument.
Step 4	webauth-vrf-aware Example: <pre>Switch (config-params-parameter-map)# webauth-vrf-aware</pre>	Enables the web-based authentication VRF aware feature on SVI.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Removing Web-Based Authentication Cache Entries

Follow these steps to remove web-based authentication cache entries:

SUMMARY STEPS

1. **enable**
2. **clear ip auth-proxy cache** *{* | host ip address}*
3. **clear ip admission cache** *{* | host ip address}*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	clear ip auth-proxy cache <i>{* host ip address}</i> Example: Switch# clear ip auth-proxy cache 192.168.4.5	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.
Step 3	clear ip admission cache <i>{* host ip address}</i> Example: Switch# clear ip admission cache 192.168.4.5	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.

Verifying Web-Based Authentication Status

Use the commands in this topic to display the web-based authentication settings for all interfaces or for specific ports.

Table 148: Privileged EXEC show Commands

Command	Purpose
show authentication sessions method webauth	Displays the web-based authentication settings for all interfaces for fastethernet, gigabitethernet, or tengigabitethernet
show wireless client mac-address a.a.a detail	Displays the session specific wireless information and wireless states.

Command	Purpose
show authentication sessions interface <i>type slot/port[details]</i>	Displays the web-based authentication settings for the specified interface for fastethernet, gigabitethernet, or tengigabitethernet. In Session Aware Networking mode, use the show access-session interface command.



CHAPTER 71

Auto Identity

The Auto Identity feature provides a set of built-in policies at global configuration and interface configuration modes. This feature is available only in Class-Based Policy Language (CPL) control policy-equivalent new-style mode. To convert all the relevant authentication commands to their CPL control policy-equivalents, use the **authentication convert-to new-style** command.

This module describes the feature and explains how to configure it.

- [Information About Auto Identity, on page 1477](#)
- [How to Configure Auto Identity, on page 1481](#)
- [Configuration Examples for Auto Identity, on page 1483](#)
- [Verifying Auto Identity, on page 1484](#)
- [Feature Information for Auto Identity, on page 1487](#)

Information About Auto Identity

Auto Identity Overview

The Cisco Identity-Based Networking Services (IBNS) solution provides a policy and identity-based framework in which edge devices can deliver flexible and scalable services to subscribers. IBNS allows the concurrent operation of IEEE 802.1x (dot1x), MAC authentication bypass (MAB), and web authentication methods, making it possible to invoke multiple authentication methods in parallel, on a single subscriber session. These authentication methods, dot1x, authentication, authorization, and accounting (AAA), and RADIUS are available in global configuration and interface configuration modes.

The Auto Identity feature uses the Cisco Common Classification Policy Language-based configuration that significantly reduces the number of commands used to configure both authentication methods and interface-level commands. The Auto Identity feature provides a set of built-in policies that are based on policy maps, class maps, parameter maps, and interface templates.

In global configuration mode, the **source template AI_GLOBAL_CONFIG_TEMPLATE** command enables the Auto Identity feature. In interface configuration mode, configure the **AI_MONITOR_MODE**, **AI_LOW_IMPACT_MODE**, or **AI_CLOSED_MODE** interface templates to enable the feature on interfaces.

You can configure multiple templates; however, you must bind multiple templates together using the **merge** command. If you do not bind the templates, the last configured template is used. While binding templates, if the same command is repeated in two templates with different arguments, the last configured command is used.



Note You can also enable user-defined templates that are configured using the **template name** command in global configuration mode .

Use the **show template interface** or **show template global** commands to display information about built-in templates. Built-in templates can be edited. Built-in template information is displayed in the output of the **show running-config** command, if the template is edited. If you delete an edited built-in template, the built-in template reverts to the default and is not deleted from the configuration. However, if you delete a user-defined template, it is deleted from the configuration.



Note Before you delete a template, ensure that it is not attached to a device.

Auto Identity Global Template

To enable the global template, configure the **source template template-name** command.



Note You must configure the RADIUS server commands, because these are not automatically configured when the global template is enabled.

The following example shows how to enable the global template:

```
Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE
Switch(config)# radius server ISE
Switch(config-radius-server)# address ipv4 172.20.254.4 auth-port 1645 acct-port 1646
Switch(config-radius-server)# key cisco
Switch(config-radius-server)# end
```

The AI_GLOBAL_CONFIG_TEMPLATE automatically configures the following commands:

```
dot1x system-auth-control
aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
```

Auto Identity Interface Templates

The following interface templates are available in the Auto Identity feature:

- AI_MONITOR_MODE—Passively monitors sessions that have authentication in open mode.

- **AI_LOW_IMPACT_MODE**—Similar to monitor mode, but with a configured static policy such as a port access control list (PACL).
- **AI_CLOSED_MODE**—Secure mode in which data traffic is not allowed into the network, until authentication is complete. This mode is the default.



Note Multi-auth host mode is not supported with the LAN Lite license.

The following commands are inbuilt in the **AI_MONITOR_MODE**:

```
switchport mode access
access-session port-control auto
access-session host-mode multi-auth
dot1x pae authenticator
mab
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

The following commands are inbuilt in the **AI_LOW_IMPACT_MODE**:

```
switchport mode access
access-session port-control auto
access-session host-mode multi-auth
dot1x pae authenticator
mab
ip access-group AI_PORT_ACL in
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

The following commands are inbuilt in the **AI_CLOSED_MODE**:

```
switchport mode access
access-session closed
access-session port-control auto
access-session host-mode multi-auth
dot1x pae authenticator
mab
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

Auto Identity Built-in Policies

The following five built-in policies are available in the Auto Identity feature:

- **AI_DOT1X_MAB_AUTH**—Enables flexible authentication with dot1x, and then MAC Address Bypass (MAB).
- **AI_DOT1X_MAB_POLICIES**—Enables flexible authentication with dot1x, and then MAB. Applies critical VLAN in case the Authentication, Authorization, and Accounting (AAA) server is not reachable.
- **AI_DOT1X_MAB_WEBAUTH**—Enables flexible authentication with dot1x, MAB, and then web authentication.
- **AI_NEXTGEN_AUTHBYBASS**—Skips authentication if an IP phone device is detected. Enables the **device classifier** command in global configuration mode and the **voice-vlan** command in interface configuration mode to detect the device. This is a reference policy map, and users can copy the contents of this policy map to other policy maps.
- **AI_STANDALONE_WEBAUTH**—Defines standalone web authentication.

Auto Identity Class Maps Templates

The following built-in class maps are supported by the Auto Identity feature:

- `AI_NRH`—Specifies that the nonresponsive host (NRH) authentication method is enabled.
- `AI_WEBAUTH_METHOD`—Specifies that the web authentication method is enabled.
- `AI_WEBAUTH_FAILED`—Specifies that the web authentication method failed to authenticate.
- `AI_WEBAUTH_NO_RESP`—Specifies that the web authentication client failed to respond.
- `AI_DOT1X_METHOD`—Specifies that the dot1x method is enabled.
- `AI_DOT1X_FAILED`—Specifies that the dot1x method failed to authenticate.
- `AI_DOT1X_NO_RESP`—Specifies that the dot1x client failed to respond.
- `AI_DOT1X_TIMEOUT`—Specifies that the dot1x client stopped responding after the initial acknowledge (ACK) request.
- `AI_MAB_METHOD`—Specifies that the MAC Authentication Bypass (MAB) method is enabled.
- `AI_MAB_FAILED`—Specifies that the MAB method failed to authenticate.
- `AI_AAA_SVR_DOWN_AUTHD_HOST`—Specifies that the Authentication, Authorization, and Accounting (AAA) server is down, and the client is in authorized state.
- `AI_AAA_SVR_DOWN_UNAUTHD_HOST`—Specifies that the AAA server is down, and the client is in authorized state.
- `AI_IN_CRITICAL_AUTH`—Specifies that the critical authentication service template is applied.
- `AI_NOT_IN_CRITICAL_AUTH`—Specifies that the critical authentication service template is not applied.
- `AI_METHOD_DOT1X_DEVICE_PHONE`—Specifies that the method is dot1x and the device type is IP phone.
- `AI_DEVICE_PHONE`—Specifies that the device type is IP phone.

Auto Identity Parameter Maps

The following built-in parameter map templates are supported by the Auto Identity feature:

- `AI_NRH_PMAP`—Starts nonresponsive host (NRH) authentication.
- `AI_WEBAUTH_PMAP`—Starts web authentication.

Auto Identity Service Templates

Service templates are available inside built-in policy maps. The following built-in service templates are supported by the Auto Identity feature:

- `AI_INACTIVE_TIMER`—Template to start the inactivity timer.
- `AI_CRITICAL_ACL`—Dummy template; users can configure this template as per their requirements.

How to Configure Auto Identity

Configuring Auto Identity Globally

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `sourcetemplate {AI_GLOBAL_CONFIG_TEMPLATE | template-name}`
4. `aaa new-model`
5. `radius server name`
6. `address ipv4 {hostname | ipv4-address}`
7. `key ipv4 {0 string | 7 string} string`
8. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	sourcetemplate {AI_GLOBAL_CONFIG_TEMPLATE <i>template-name</i>} Example: Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE	Configures an auto identity template. <ul style="list-style-type: none"> • AI_GLOBAL_CONFIG_TEMPLATE is a built-in template. • <i>template-name</i> is a user-defined template.
Step 4	aaa new-model Example: Switch(config)# aaa new-model	Enables the authentication, authorization, and accounting (AAA) access control mode.
Step 5	radius server <i>name</i> Example: Switch(config)# radius server ISE	Specifies the name for the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode.
Step 6	address ipv4 {<i>hostname</i> <i>ipv4-address</i>} Example: Switch(config-radius-server)# address ipv4 10.1.1.1	Configures the IPv4 address for the RADIUS server accounting and authentication parameters. <p>Note This command is not a part of the global template, and you must configure it.</p>

	Command or Action	Purpose
Step 7	key ipv4 {0 string 7 string} string Example: Switch(config-radius-server)# key ipv4 cisco	Specifies the authentication and encryption key for all RADIUS communications between the device and the RADIUS server. Note This command is not a part of the global template, and you must configure it.
Step 8	end Example: Switch(config-radius-server)# end	Exits RADIUS server configuration mode and returns to privileged EXEC mode.

Configuring Auto Identity at an Interface Level

When you configure two interface templates, you must configure the **merge** keyword. If you do not, the last configured template is used.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **source template** {AI_CLOSED_MODE | AI_LOW_IMPACT_MODE | AI_MONITOR_MODE | *template-name*} [**merge**]
5. **source template** {AI_CLOSED_MODE | AI_LOW_IMPACT_MODE | AI_MONITOR_MODE | *template-name*} [**merge**]
6. **switchport access vlan** *vlan-id*
7. **switchport voice vlan** *vlan-id*
8. Repeat Steps 4, 6, and 7 on all interfaces that must have the Auto Identity feature configured.
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Configures an interface and enters interface configuration mode.

	Command or Action	Purpose
Step 4	source template {AI_CLOSED_MODE AI_LOW_IMPACT_MODE AI_MONITOR_MODE <i>template-name</i> } [merge] Example: Switch(config-if)# source template AI_CLOSED_MODE	Configures a source template for the interface.
Step 5	source template {AI_CLOSED_MODE AI_LOW_IMPACT_MODE AI_MONITOR_MODE <i>template-name</i> } [merge] Example: Switch(config-if)# source template AI_MONITOR_MODE merge	(Optional) Configures a source template for the interface and merges this template with the previously configured template <ul style="list-style-type: none"> When you configure two templates, if you do not configure the merge keyword, the last configured template is used.
Step 6	switchport access vlan <i>vlan-id</i> Example: Switch(config-if)# switchport access vlan 100	Sets the VLAN when the interface is in access mode.
Step 7	switchport voice vlan <i>vlan-id</i> Example: Switch(config-if)# switchport voice vlan 101	Configures a voice VLAN on a multiple VLAN access port.
Step 8	Repeat Steps 4, 6, and 7 on all interfaces that must have the Auto Identity feature configured.	—
Step 9	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for Auto Identity

Example: Configuring Auto Identity Globally

```
Switch> enable
Switch# configure terminal
Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE
Switch(config)# aaa new-model
Switch(config)# radius server ISE
Switch(config-radius-server)# address ipv4 10.1.1.1
Switch(config-radius-server)# key ipv4 cisco
Switch(config-radius-server)# end
```

Example: Configuring Auto Identity at an Interface Level

```
Switch> enable
Switch# configure terminal
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# source template AI_CLOSED_MODE
Switch(config-if)# source template AI_MONITOR_MODE merge
Switch(config-if)# switchport access vlan 100
Switch(config-if)# switchport voice vlan 101
Switch(config-if)# end
```

Verifying Auto Identity

Step 1 enable

Example:

```
Switch> enable
```

Enables Privileged EXEC mode.

- Enter your password if prompted.

Step 2 show template interface source built-in all

Displays all the configured built-in interface templates.

Example:

```
Switch# show template interface source built-in all

Template Name      : AI_CLOSED_MODE
Modified           : No
Template Definition :
dot1x pae authenticator
switchport mode access
mab
access-session closed
access-session port-control auto
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
!
Template Name      : AI_LOW_IMPACT_MODE
Modified           : No
Template Definition :
dot1x pae authenticator
switchport mode access
mab
access-session port-control auto
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
ip access-group AI_PORT_ACL in
!
Template Name      : AI_MONITOR_MODE
Modified           : No
Template Definition :
dot1x pae authenticator
switchport mode access
mab
access-session port-control auto
```

```
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
!
```

Step 3 show template global source built-in all

Displays all the configured global built-in templates.

Example:

```
Switch# show template global source built-in all

Global Template Name      : AI_GLOBAL_CONFIG_TEMPLATE
Modified                  : No
Global Template Definition : global
dot1x system-auth-control
aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
!
```

Step 4 show derived-config | include aaa | radius-server

Displays the composite results of all the configuration commands that apply to an interface, including commands that come from sources such as static templates, dynamic templates, dialer interfaces, and authentication, authorization, and accounting (AAA) per-user attributes.

Example:

```
Switch# show derived-config | inc aaa | radius-server

aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
aaa session-id common
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
radius-server host 10.25.18.42 key cisco
```

Step 5 show derived-config | interface type-number

Displays the composite results of all configuration for an interface.

Example:

```
Switch# show derived-config | interface gigabitethernet2/0/6

Building configuration...
```

```

Derived configuration : 267 bytes
!
interface GigabitEthernet2/0/6
  switchport mode access
  switchport voice vlan 100
  access-session closed
  access-session port-control auto
  mab
  dot1x pae authenticator
  spanning-tree portfast edge
  service-policy type control subscriber AI_DOT1X_MAB_POLICIES
end

```

Step 6 **show access-session | interface *interface-type-number* details**

Displays the policies applied to an interface.

Example:

```
Switch# show access-session interface gigabitethernet2/0/6 details
```

```

Interface: GigabitEthernet2/0/6
  MAC Address: c025.5c43.be00
  IPv6 Address: Unknown
  IPv4 Address: Unknown
  User-Name: CP-9971-SEPC0255C43BE00
  Device-type: Cisco-IP-Phone-9971
  Status: Authorized
  Domain: VOICE
  Oper host mode: multi-auth
  Oper control dir: both
  Session timeout: N/A
  Common Session ID: 091A1C5B00000017002003EE
  Acct Session ID: 0x00000005
  Handle: 0xBB00000B
  Current Policy: AI_DOT1X_MAB_POLICIES

```

Local Policies:

Server Policies:

```

  Vlan Group: Vlan: 100
  Security Policy: Must Not Secure
  Security Status: Link Unsecure

```

Method status list:

Method	State
dot1x	Authc Success

Step 7 **show running-config interface *type-number***

Displays the contents of the current running configuration file or the configuration for an interface.

Example:

```
Switch# show running-config interface gigabitethernet2/0/6
```

Building configuration...

Current configuration : 214 bytes

```

!
interface GigabitEthernet2/0/6
 switchport mode access
 switchport voice vlan 100
 access-session port-control auto
 spanning-tree portfast edge
 service-policy type control subscriber AI_NEXTGEN_AUTHBYPASS
end

```

Step 8 show lldp neighbor

Displays information about one or all neighboring devices discovered using the Link Layer Discovery Protocol (LLDP).

Example:

```
Switch# show lldp neighbor
```

Capability codes:

```

(R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
(W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other

```

```

Device ID           Local Intf      Hold-time  Capability      Port ID
SEPC0255C43BE00    Gi2/0/6        180        B,T             C0255C43BE00:P1

```

Total entries displayed: 1

Feature Information for Auto Identity

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 149: Feature Information for Auto Identity

Feature Name	Releases	Feature Information
Auto Identity	Cisco IOS Release 15.2(4)E	<p>The Auto Identity feature provides a set of built-in policies at the global configuration and interface configuration modes. This feature is available only in the Class-Based Policy Language (CPL) control policy-equivalent new-style mode.</p> <p>In Cisco IOS Release 15.2(4)E, this feature was implemented on Cisco Catalyst 2960–X Series Switches, Catalyst 3750–X Series Switches, and Cisco Catalyst 4500E Supervisor Engine 7-E.</p> <p>The following commands was introduced or modified: source-template.</p>



CHAPTER 72

Configuring Port-Based Traffic Control

- Finding Feature Information, on page 1489
- Information About Storm Control, on page 1490
- How to Configure Storm Control, on page 1491
- Information About Protected Ports, on page 1496
- How to Configure Protected Ports, on page 1497
- Monitoring Protected Ports, on page 1498
- Where to Go Next, on page 1499
- Information About Port Blocking, on page 1499
- How to Configure Port Blocking, on page 1499
- Monitoring Port Blocking, on page 1501
- Prerequisites for Port Security, on page 1501
- Restrictions for Port Security, on page 1501
- Information About Port Security, on page 1502
- How to Configure Port Security, on page 1506
- Configuration Examples for Port Security, on page 1512
- Information About Protocol Storm Protection, on page 1513
- How to Configure Protocol Storm Protection, on page 1514
- Monitoring Protocol Storm Protection, on page 1515

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Information About Storm Control

Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation, mistakes in network configurations, or users issuing a denial-of-service attack can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

How Traffic Activity is Measured

Storm control uses one of these methods to measure traffic activity:

- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received

With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.

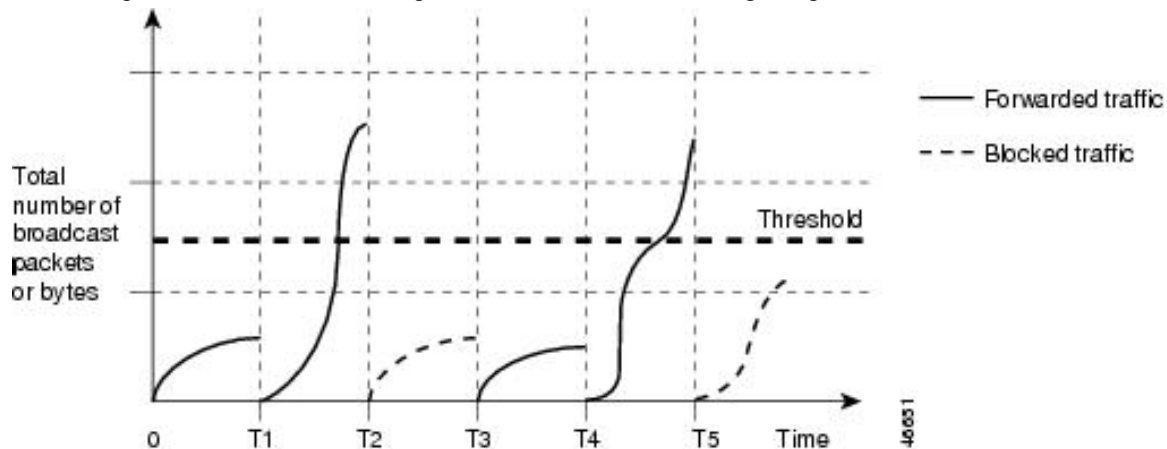


Note When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BPDU) and Cisco Discovery Protocol frames, are blocked. However, the switch does not differentiate between routing updates, such as OSPF, and regular multicast data traffic, so both types of traffic are blocked.

Traffic Patterns

Figure 119: Broadcast Storm Control Example

This example shows broadcast traffic patterns on an interface over a given period of time.



Broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

The combination of the storm-control suppression level and the 1-second time interval controls the way the storm control algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic. A value of 0.0 means that all broadcast, multicast, or unicast traffic on that port is blocked.



Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

How to Configure Storm Control

Configuring Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used for a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.



Note Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

Follow these steps to storm control and threshold levels:

Before you begin

Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}
5. **storm-control action** {**shutdown** | **trap**}
6. **end**
7. **show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	storm-control { broadcast multicast unicast } level { <i>level</i> [<i>level-low</i>] bps <i>bps</i> [<i>bps-low</i>] pps <i>pps</i> [<i>pps-low</i>]} Example:	Configures broadcast, multicast, or unicast storm control. By default, storm control is disabled. The keywords have these meanings:

	Command or Action	Purpose
	<pre>Switch(config-if)# storm-control unicast level 87 65</pre>	<ul style="list-style-type: none"> • For <i>level</i>, specifies the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is 0.00 to 100.00. • (Optional) For <i>level-low</i>, specifies the falling threshold level as a percentage (up to two decimal places) of the bandwidth. This value must be less than or equal to the rising suppression value. The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is 0.00 to 100.00. If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0.0, all broadcast, multicast, and unicast traffic on that port is blocked. • For bps <i>bps</i>, specifies the rising threshold level for broadcast, multicast, or unicast traffic in bits per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For <i>bps-low</i>, specifies the falling threshold level in bits per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. • For pps <i>pps</i>, specifies the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For <i>pps-low</i>, specifies the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. <p>For BPS and PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.</p>
Step 5	<p>storm-control action {shutdown trap}</p> <p>Example:</p> <pre>Switch(config-if)# storm-control action trap</pre>	<p>Specifies the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps.</p> <ul style="list-style-type: none"> • Select the shutdown keyword to error-disable the port during a storm.

	Command or Action	Purpose
		<ul style="list-style-type: none"> Select the trap keyword to generate an SNMP trap when a storm is detected.
Step 6	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 7	show storm-control [<i>interface-id</i>] [broadcast multicast unicast] Example: <pre>Switch# show storm-control gigabitethernet1/0/1 unicast</pre>	Verifies the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, details for all traffic types (broadcast, multicast and unicast) are displayed.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Small-Frame Arrival Rate

Incoming VLAN-tagged packets smaller than 67 bytes are considered small frames. They are forwarded by the switch, but they do not cause the switch storm-control counters to increment.

You globally enable the small-frame arrival feature on the switch and then configure the small-frame threshold for packets on each interface. Packets smaller than the minimum size and arriving at a specified rate (the threshold) are dropped since the port is error disabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **errdisable detect cause small-frame**
4. **errdisable recovery interval** *interval*
5. **errdisable recovery cause small-frame**
6. **interface** *interface-id*
7. **small-frame violation-rate** *pps*
8. **end**
9. **show interfaces** *interface-id*
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	errdisable detect cause small-frame Example: <pre>Switch(config)# errdisable detect cause small-frame</pre>	Enables the small-frame rate-arrival feature on the switch.
Step 4	errdisable recovery interval <i>interval</i> Example: <pre>Switch(config)# errdisable recovery interval 60</pre>	(Optional) Specifies the time to recover from the specified error-disabled state.
Step 5	errdisable recovery cause small-frame Example: <pre>Switch(config)# errdisable recovery cause small-frame</pre>	(Optional) Configures the recovery time for error-disabled ports to be automatically re-enabled after they are error disabled by the arrival of small frames Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.
Step 6	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/2</pre>	Enters interface configuration mode, and specify the interface to be configured.
Step 7	small-frame violation-rate <i>pps</i> Example: <pre>Switch(config-if)# small-frame violation rate 10000</pre>	Configures the threshold rate for the interface to drop incoming packets and error disable the port. The range is 1 to 10,000 packets per second (pps)

	Command or Action	Purpose
Step 8	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	show interfaces <i>interface-id</i> Example: <pre>Switch# show interfaces gigabitethernet1/0/2</pre>	Verifies the configuration.
Step 10	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 11	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Information About Protected Ports

Protected Ports

Some applications require that no traffic be forwarded at Layer 2 between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.

Because a switch stack represents a single logical switch, Layer 2 traffic is not forwarded between any protected ports in the switch stack, whether they are on the same or different switches in the stack.

Default Protected Port Configuration

The default is to have no protected ports defined.

Protected Ports Guidelines

You can configure protected ports on a physical interface (for example, Gigabit Ethernet port 1) or an EtherChannel group (for example, port-channel 5). When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

How to Configure Protected Ports

Configuring a Protected Port

Before you begin

Protected ports are not pre-defined. This is the task to configure one.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport protected**
5. **end**
6. **show interfaces *interface-id* switchport**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example:	Specifies the interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Switch(config) # <code>interface gigabitethernet 1/0/1</code>	
Step 4	switchport protected Example: Switch(config-if) # <code>switchport protected</code>	Configures the interface to be a protected port.
Step 5	end Example: Switch(config) # <code>end</code>	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: Switch# <code>show interfaces gigabitethernet 1/0/1 switchport</code>	Verifies your entries.
Step 7	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries.
Step 8	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring Protected Ports

Table 150: Commands for Displaying Protected Port Settings

Command	Purpose
<code>show interfaces [<i>interface-id</i>] switchport</code>	Displays the administrative and operational status of all switch (nonrouting) ports or the specified port, including port blocking protection settings.

Where to Go Next

Information About Port Blocking

Port Blocking

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or nonprotected) from flooding unknown unicast or multicast packets to other ports.



Note With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

How to Configure Port Blocking

Blocking Flooded Traffic on an Interface

Before you begin

The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port-channel group.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport block multicast**
5. **switchport block unicast**
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	switchport block multicast Example: Switch(config-if)# switchport block multicast	Blocks unknown multicast forwarding out of the port.
Step 5	switchport block unicast Example: Switch(config-if)# switchport block unicast	Blocks unknown unicast forwarding out of the port.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show interfaces <i>interface-id</i> switchport Example: Switch# show interfaces gigabitethernet 1/0/1 switchport	Verifies your entries.
Step 8	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# <code>show running-config</code>	
Step 9	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring Port Blocking

Table 151: Commands for Displaying Port Blocking Settings

Command	Purpose
<code>show interfaces [interface-id] switchport</code>	Displays the administrative and operational status of all sw (nonrouting) ports or the specified port, including port bloc protection settings.

Prerequisites for Port Security



Note If you try to set the maximum value to a number less than the number of secure addresses already configured on an interface, the command is rejected.

Restrictions for Port Security

- The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.

Information About Port Security

Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address, the workstation attached to that port is assured the full bandwidth of the port.

If a port is configured as a secure port and the maximum number of secure MAC addresses is reached, when the MAC address of a station attempting to access the port is different from any of the identified secure MAC addresses, a security violation occurs. Also, if a station with a secure MAC address configured or learned on one secure port attempts to access another secure port, a violation is flagged.

Types of Secure MAC Addresses

The switch supports these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address *mac-address*** interface configuration command, stored in the address table, and added to the switch running configuration.
- Dynamic secure MAC addresses—These are dynamically configured, stored only in the address table, and removed when the switch restarts.
- Sticky secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, when the switch restarts, the interface does not need to dynamically reconfigure them.

Sticky Secure MAC Addresses

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling sticky learning. The interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.
- Running diagnostic tests with port security enabled.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- **protect**—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.



Note We do not recommend configuring the protect violation mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- **restrict**—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.
- **shutdown**—a port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shut down** interface configuration commands. This is the default mode.
- **shutdown vlan**—Use to set the security violation mode per-VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs

This table shows the violation mode and the actions taken when you configure an interface for port security.

Table 152: Security Violation Mode Actions

Violation Mode	Traffic is forwarded 19	Sends SNMP trap	Sends syslog message	Displays error message 20	Violation counter increments	Shut down
protect	No	No	No	No	No	No
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	No	No	No	Yes	Yes
shutdown vlan	No	No	Yes	No	Yes	No 21

- ¹⁹ Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.
- ²⁰ The switch returns an error message if you manually configure an address that would cause a security violation.
- ²¹ Shuts down only the VLAN on which the violation occurred.

Port Security Aging

You can use port security aging to set the aging time for all secure addresses on a port. Two types of aging are supported per port:

- Absolute—The secure addresses on the port are deleted after the specified aging time.
- Inactivity—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Default Port Security Configuration

Table 153: Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.
Maximum number of secure MAC addresses per port	1.
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

- Port security can only be configured on static access ports or trunk ports. A secure port cannot be a dynamic access port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.
- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC

addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the phone.

- When a trunk port configured with port security and assigned to an access VLAN for data traffic and to a voice VLAN for voice traffic, entering the **switchport voice** and **switchport priority extend** interface configuration commands has no effect.

When a connected device uses the same MAC address to request an IP address for the access VLAN and then an IP address for the voice VLAN, only the access VLAN is assigned an IP address.

- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.
- The switch does not support port security aging of sticky secure MAC addresses.

This table summarizes port security compatibility with other port-based features.

Table 154: Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
DTP ²² port ²³	No
Trunk port	Yes
Dynamic-access port ²⁴	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	Yes
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port ²⁵	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

²² DTP=Dynamic Trunking Protocol

²³ A port configured with the **switchport mode dynamic** interface configuration command.

²⁴ A VLAN Query Protocol (VQP) port configured with the **switchport access vlan dynamic** interface configuration command.

- ²⁵ You must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN.

How to Configure Port Security

Enabling and Configuring Port Security

Before you begin

This task restricts input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode** {**access** | **trunk**}
5. **switchport voice vlan** *vlan-id*
6. **switchport port-security**
7. **switchport port-security** [**maximum value** [**vlan** {*vlan-list* | {**access** | **voice**}}]]
8. **switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}
9. **switchport port-security** [**mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]]
10. **switchport port-security mac-address sticky**
11. **switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}]]
12. **end**
13. **show port-security**
14. **show running-config**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/1</pre>	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	switchport mode {access trunk} Example: <pre>Switch(config-if)# switchport mode access</pre>	Sets the interface switchport mode as access or trunk; an interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 5	switchport voice vlan <i>vlan-id</i> Example: <pre>Switch(config-if)# switchport voice vlan 22</pre>	Enables voice VLAN on a port. vlan-id—Specifies the VLAN to be used for voice traffic.
Step 6	switchport port-security Example: <pre>Switch(config-if)# switchport port-security</pre>	Enable port security on the interface. Note Under certain conditions, when port security is enabled on the member ports in a switch stack, the DHCP and ARP packets would be dropped. To resolve this, configure a shut and no shut on the interface.
Step 7	switchport port-security [maximum <i>value</i> [vlan { <i>vlan-list</i> {access voice}}]] Example: <pre>Switch(config-if)# switchport port-security maximum 20</pre>	<p>(Optional) Sets the maximum number of secure MAC addresses for the interface. The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is set by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.</p> <p>(Optional) vlan—sets a per-VLAN maximum value</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • <i>vlan-list</i>—On a trunk port, you can set a per-VLAN maximum value on a range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. • access—On an access port, specifies the VLAN as an access VLAN.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.</p>
Step 8	<p>switchport port-security violation {protect restrict shutdown shutdown vlan}</p> <p>Example:</p> <pre>Switch(config-if)# switchport port-security violation restrict</pre>	<p>(Optional) Sets the violation mode, the action to be taken when a security violation is detected, as one of these:</p> <ul style="list-style-type: none"> • protect—When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. <p>Note We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.</p> <ul style="list-style-type: none"> • restrict—When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • shutdown—The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • shutdown vlan—Use to set the security violation mode per VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs.

	Command or Action	Purpose
		<p>Note When a secure port is in the error-disabled state, you can bring it out of this state by entering the errdisable recovery cause psecure-violation global configuration command. You can manually re-enable it by entering the shutdown and no shutdown interface configuration commands or by using the clear errdisable interface vlan privileged EXEC command.</p>
<p>Step 9</p>	<p>switchport port-security [<i>mac-address mac-address</i> [<i>vlan {vlan-id} {access voice}</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice</pre>	<p>(Optional) Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned.</p> <p>Note If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration.</p> <p>(Optional) vlan—sets a per-VLAN maximum value.</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • <i>vlan-id</i>—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.</p>
<p>Step 10</p>	<p>switchport port-security mac-address sticky</p> <p>Example:</p> <pre>Switch(config-if)# switchport port-security mac-address sticky</pre>	<p>(Optional) Enables sticky learning on the interface.</p>

	Command or Action	Purpose
Step 11	<p>switchport port-security mac-address sticky <code>[mac-address vlan {vlan-id {access voice}}]</code></p> <p>Example:</p> <pre>Switch(config-if)# switchport port-security mac-address sticky 00:A0:C7:12:C9:25 vlan voice</pre>	<p>(Optional) Enters a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration.</p> <p>Note If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address.</p> <p>(Optional) vlan—sets a per-VLAN maximum value. Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • vlan-id—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN.</p>
Step 12	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 13	<p>show port-security</p> <p>Example:</p> <pre>Switch# show port-security</pre>	Verifies your entries.
Step 14	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 15	<p>copy running-config startup-config</p> <p>Example:</p>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Enabling and Configuring Port Security Aging

Use this feature to remove and add devices on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of secure addresses on a per-port basis.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `switchport port-security aging {static | time time | type {absolute | inactivity}}`
5. `end`
6. `show port-security [interface interface-id] [address]`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>interface interface-id</code></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet 1/0/1</pre>	<p>Specifies the interface to be configured, and enter interface configuration mode.</p>
Step 4	<p><code>switchport port-security aging {static time time type {absolute inactivity}}</code></p> <p>Example:</p> <pre>Switch(config-if)# switchport port-security aging</pre>	<p>Enables or disable static aging for the secure port, or set the aging time or type.</p> <p>Note The switch does not support port security aging of sticky secure addresses.</p>

	Command or Action	Purpose
	<code>time 120</code>	<p>Enter static to enable aging for statically configured secure addresses on this port.</p> <p>For <i>time</i>, specifies the aging time for this port. The valid range is from 0 to 1440 minutes.</p> <p>For type, select one of these keywords:</p> <ul style="list-style-type: none"> • absolute—Sets the aging type as absolute aging. All the secure addresses on this port age out exactly after the time (minutes) specified lapses and are removed from the secure address list. • inactivity—Sets the aging type as inactivity aging. The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show port-security [interface <i>interface-id</i>] [address]</p> <p>Example:</p> <pre>Switch# show port-security interface gigabitethernet 1/0/1</pre>	Verifies your entries.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuration Examples for Port Security

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled.

```
Switch(config)# interface gigabitethernet 1/0/1
```



```
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 50
Switch(config-if)# switchport port-security mac-address sticky
```

This example shows how to configure a static secure MAC address on VLAN 3 on a port:

```
Switch(config)# interface gigabitethernet 1/0/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address 0000.0200.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Switch(config)# interface tengigabitethernet 1/0/1
Switch(config-if)# switchport access vlan 21
Switch(config-if)# switchport mode access
Switch(config-if)# switchport voice vlan 22
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 20
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Switch(config-if)# switchport port-security mac-address 0000.0000.0003
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Switch(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Switch(config-if)# switchport port-security maximum 10 vlan access
Switch(config-if)# switchport port-security maximum 10 vlan voice
```

Information About Protocol Storm Protection

Protocol Storm Protection

When a switch is flooded with Address Resolution Protocol (ARP) or control packets, high CPU utilization can cause the CPU to overload. These issues can occur:

- Routing protocol can flap because the protocol control packets are not received, and neighboring adjacencies are dropped.
- Spanning Tree Protocol (STP) reconverges because the STP bridge protocol data unit (BPDU) cannot be sent or received.
- CLI is slow or unresponsive.

Using protocol storm protection, you can control the rate at which control packets are sent to the switch by specifying the upper threshold for the packet flow rate. The supported protocols are ARP, ARP snooping, Dynamic Host Configuration Protocol (DHCP) v4, DHCP snooping, Internet Group Management Protocol (IGMP), and IGMP snooping.

When the packet rate exceeds the defined threshold, the switch drops all traffic arriving on the specified virtual port for 30 seconds. The packet rate is measured again, and protocol storm protection is again applied if necessary.

For further protection, you can manually error disable the virtual port, blocking all incoming traffic on the virtual port. You can manually enable the virtual port or set a time interval for automatic re-enabling of the virtual port.



Note Excess packets are dropped on no more than two virtual ports.

Virtual port error disabling is not supported for EtherChannel and Flexlink interfaces

Default Protocol Storm Protection Configuration

Protocol storm protection is disabled by default. When it is enabled, auto-recovery of the virtual port is disabled by default.

How to Configure Protocol Storm Protection

Enabling Protocol Storm Protection

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **psp {arp | dhcp | igmp} pps *value***
4. **errdisable detect cause psp**
5. **errdisable recovery interval *time***
6. **end**
7. **show psp config {arp | dhcp | igmp}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>psp {arp dhcp igmp} pps <i>value</i></p> <p>Example:</p> <pre>Switch(config)# psp dhcp pps 35</pre>	<p>Configures protocol storm protection for ARP, IGMP, or DHCP.</p> <p>For <i>value</i>, specifies the threshold value for the number of packets per second. If the traffic exceeds this value, protocol storm protection is enforced. The range is from 5 to 50 packets per second.</p>
Step 4	<p>errdisable detect cause psp</p> <p>Example:</p> <pre>Switch(config)# errdisable detect cause psp</pre>	<p>(Optional) Enables error-disable detection for protocol storm protection. If this feature is enabled, the virtual port is error disabled. If this feature is disabled, the port drops excess packets without error disabling the port.</p>
Step 5	<p>errdisable recovery interval <i>time</i></p> <p>Example:</p> <pre>Switch</pre>	<p>(Optional) Configures an auto-recovery time (in seconds) for error-disabled virtual ports. When a virtual port is error-disabled, the switch auto-recovers after this time. The range is from 30 to 86400 seconds.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show psp config {arp dhcp igmp}</p> <p>Example:</p> <pre>Switch# show psp config dhcp</pre>	<p>Verifies your entries.</p>

Monitoring Protocol Storm Protection

Command	Purpose
show psp config {arp dhcp igmp}	Verify your entries.



CHAPTER 73

Configuring IPv6 First Hop Security

- [Finding Feature Information](#), on page 1517
- [Prerequisites for First Hop Security in IPv6](#), on page 1517
- [Restrictions for First Hop Security in IPv6](#), on page 1517
- [Information about First Hop Security in IPv6](#), on page 1518
- [How to Configure an IPv6 Snooping Policy](#), on page 1520
- [How to Configure the IPv6 Binding Table Content](#), on page 1525
- [How to Configure an IPv6 Neighbor Discovery Inspection Policy](#), on page 1526
- [How to Configure an IPv6 Router Advertisement Guard Policy](#), on page 1531
- [How to Configure an IPv6 DHCP Guard Policy](#), on page 1536
- [How to Configure IPv6 Source Guard](#), on page 1542
- [How to Configure IPv6 Source Guard](#), on page 1544
- [How to Configure IPv6 Prefix Guard](#), on page 1547

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.
- QoS should be enabled on the switch before configuring CoPP policies using **mls qos** command.

Restrictions for First Hop Security in IPv6

- The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):

- A physical port with an FHS policy attached cannot join an EtherChannel group.
- An FHS policy cannot be attached to a physical port when it is a member of an EtherChannel group.
- By default, a snooping policy has a security-level of guard. When such a snooping policy is configured on an access switch, external IPv6 Router Advertisement (RA) or Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server packets are blocked, even though the uplink port facing the router or DHCP server/relay is configured as a trusted port. To allow IPv6 RA or DHCPv6 server messages, do the following:
 - Apply an IPv6 RA-guard policy (for RA) or IPv6 DHCP-guard policy (for DHCP server messages) on the uplink port.
 - Configure a snooping policy with a lower security-level, for example glean or inspect. However, configuring a lower security level is not recommended with such a snooping policy, because benefits of First Hop security features are not effective.
- The following restrictions apply for CoPP policies with IPv6 SISF-based device tracking policies due to limitation reported in [CSCvk32439](#):
 - CoPP policies are required to limit IPv6 NDP traffic when IPv6 SISF policies are configured on the switch.
 - After NDP CoPP policies are configured, limited traffic hits CPU. To accommodate the total end points connected, the number of NDP CoPP policies should be slightly more than the number of users connected to each switch in a stack. If you configure NDP CoPP policies less than the number of end points connected to the switch, the IP allocation to the end point is delayed but is not ignored completely.



Note For example, if a stack of 5 switches has approximately 300 users, the NDP CoPP policies should be more than 300.

- The DHCPv6 (server-to-client and client-to-server) CoPP policies are required only if Lightweight DHCPv6 Relay Agent (LDRA) is configured under IPv6 SISF-based device tracking policies on the switch.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.
- IPv6 FHS Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer

address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.

- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

This feature mitigates some of the inherent vulnerabilities of the ND mechanism, such as attacks on DAD, address resolution, router discovery, and the neighbor cache.

- IPv6 Router Advertisement Guard—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the Layer 2 device with the information found in the received RA frame. Once the Layer 2 device has validated the content of the RA frame and router redirect frame against the configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.
- IPv6 DHCP Guard—The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.
- IPv6 Source Guard—Like IPv4 Source Guard, IPv6 Source Guard validates the source address or prefix to prevent source address spoofing.

A source guard programs the hardware to allow or deny traffic based on source or destination addresses. It deals exclusively with data packet traffic.

To debug source-guard packets, use the `debug ipv6 snooping source-guard` privileged EXEC command.

The following restrictions apply:

- An FHS policy cannot be attached to a physical port when it is a member of an EtherChannel group.
- When IPv6 source guard is enabled on a switch port, NDP or DHCP snooping must be enabled on the interface to which the switch port belongs. Otherwise, all data traffic from this port will be blocked.
- An IPv6 source guard policy cannot be attached to a VLAN. It is supported only at the interface level.
- When you configure IPv4 and IPv6 source guard together on an interface, it is recommended to use **ip verify source mac-check** instead of **ip verify source**. IPv4 connectivity on a given port might break due to two different filtering rules set — one for IPv4 (IP-filter) and the other for IPv6 (IP-MAC filter).
- You cannot use IPv6 Source Guard and Prefix Guard together. When you attach the policy to an interface, it should be "validate address" or "validate prefix" but not both.
- PVLAN and Source/Prefix Guard cannot be applied together.

For more information on IPv6 Source Guard, see the [IPv6 Source Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- **IPv6 Prefix Guard**—The IPv6 prefix guard feature works within the IPv6 source guard feature, to enable the device to deny traffic originated from non-topologically correct addresses. IPv6 prefix guard is often used when IPv6 prefixes are delegated to devices (for example, home gateways) using DHCP prefix delegation. The feature discovers ranges of addresses assigned to the link and blocks any traffic sourced with an address outside this range.

For more information on IPv6 Prefix Guard, see the [IPv6 Prefix Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- **IPv6 Destination Guard**—The IPv6 destination guard feature works with IPv6 neighbor discovery to ensure that the device performs address resolution only for those addresses that are known to be active on the link. It relies on the address glean functionality to populate all destinations active on the link into the binding table and then blocks resolutions before they happen when the destination is not found in the binding table.

For more information about IPv6 Destination Guard, see the [IPv6 Destination Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- **IPv6 Neighbor Discovery Multicast Suppress**—The IPv6 Neighbor Discovery multicast suppress feature is an IPv6 snooping feature that runs on a switch or a wireless controller and is used to reduce the amount of control traffic necessary for proper link operations.
- **DHCPv6 Relay—Lightweight DHCPv6 Relay Agent**—The DHCPv6 Relay—Lightweight DHCPv6 Relay Agent feature allows relay agent information to be inserted by an access node that performs a link-layer bridging (non-routing) function. Lightweight DHCPv6 Relay Agent (LDRA) functionality can be implemented in existing access nodes, such as DSL access multiplexers (DSLAMs) and Ethernet switches, that do not support IPv6 control or routing functions. LDRA is used to insert relay-agent options in DHCP version 6 (DHCPv6) message exchanges primarily to identify client-facing interfaces. LDRA functionality can be enabled on an interface and on a VLAN.



Note If an LDRA device is directly connected to a client, the interface must have the pool configuration to fetch the specific subnet or link information at the server side. In this case, if the LDRA device is present in different subnets or links, the server may not be able to fetch the correct subnet. You can now configure the pool name in the interface so as to choose the proper subnet or link for the client.

For more information about DHCPv6 Relay, See the [DHCPv6 Relay—Lightweight DHCPv6 Relay Agent](#) section of the IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15.1SG.

How to Configure an IPv6 Snooping Policy

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Snooping Policy :

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 snooping policy *policy-name***

3. `{[default]|[device-role {node | switch}]|[limit address-count value]|[no]|[protocol {dhcp | ndp}]|[security-level {glean | guard | inspect}]|[tracking {disable [stale-lifetime [seconds | infinite]]|enable [reachable-lifetime [seconds | infinite]]}|[trusted-port]}`
4. `end`
5. `show ipv6 snooping policy policy-name`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <code>Switch# configure terminal</code>	Enters the global configuration mode.
Step 2	ipv6 snooping policy policy-name Example: <code>Switch(config)# ipv6 snooping policy example_policy</code>	Creates a snooping policy and enters IPv6 Snooping Policy Configuration mode.
Step 3	<code>{[default] [device-role {node switch}] [limit address-count value] [no] [protocol {dhcp ndp}] [security-level {glean guard inspect}] [tracking {disable [stale-lifetime [seconds infinite]] enable [reachable-lifetime [seconds infinite]]} [trusted-port]}</code> Example: <code>Switch(config-ipv6-snooping)# security-level inspect</code> Example: <code>Switch(config-ipv6-snooping)# trusted-port</code>	<p>Enables data address gleaning, validates messages against various criteria, specifies the security level for messages.</p> <ul style="list-style-type: none"> • (Optional) default—Sets all to default options. • (Optional) device-role {node switch}—Specifies the role of the device attached to the port. Default is node. • (Optional) limit address-count value—Limits the number of addresses allowed per target. • (Optional) no—Negates a command or sets it to defaults. • (Optional) protocol {dhcp ndp}—Specifies which protocol should be redirected to the snooping feature for analysis. The default, is dhcp and ndp. To change the default, use the no protocol command. • (Optional) security-level {glean guard inspect}—Specifies the level of security enforced by the feature. Default is guard. <ul style="list-style-type: none"> glean—Gleans addresses from messages and populates the binding table without any verification. guard—Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option. inspect—Gleans addresses, validates messages for consistency and conformance, and enforces address ownership.

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) tracking {disable enable}—Overrides the default tracking behavior and specifies a tracking option. (Optional) trusted-port—Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table.
Step 4	end Example: Switch(config-ipv6-snooping) # exit	Exits configuration modes to Privileged EXEC mode.
Step 5	show ipv6 snooping policy <i>policy-name</i> Example: Switch# show ipv6 snooping policy example_policy	Displays the snooping policy configuration.

What to do next

Attach an IPv6 Snooping policy to interfaces or VLANs.

How to Attach an IPv6 Snooping Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **switchport**
4. **ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
5. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type <i>stack/module/port</i> Example:	Specifies an interface type and identifier; enters the interface configuration mode.

	Command or Action	Purpose
	Switch(config)# interface gigabitethernet 1/1/4	
Step 3	<p>switchport</p> <p>Example:</p> <pre>Switch(config-if)# switchport</pre>	<p>Enters the Switchport mode.</p> <p>Note To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the switchport interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration. The command prompt displays as (config-if)# in Switchport configuration mode.</p>
Step 4	<p>ipv6 snooping [attach-policy policy_name [vlan {vlan_id add vlan_ids exceptvlan_ids none remove vlan_ids}] vlan {vlan_id add vlan_ids exceptvlan_ids none remove vlan_ids all}]</p> <p>Example:</p> <pre>Switch(config-if)# ipv6 snooping</pre> <p>or</p> <pre>Switch(config-if)# ipv6 snooping attach-policy example_policy</pre> <p>or</p> <pre>Switch(config-if)# ipv6 snooping vlan 111,112</pre> <p>or</p> <pre>Switch(config-if)# ipv6 snooping attach-policy example_policy vlan 111,112</pre>	<p>Attaches a custom ipv6 snooping policy to the interface or the specified VLANs on the interface. To attach the default policy to the interface, use the ipv6 snooping command without the attach-policy keyword. To attach the default policy to VLANs on the interface, use the ipv6 snooping vlan command. The default policy is, security-level guard, device-role node, protocol ndp and dhcp.</p>
Step 5	<p>do show running-config</p> <p>Example:</p> <pre>Switch#(config-if)# do show running-config</pre>	<p>Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.</p>

How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Switch(config)# interface range Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 snooping [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Switch(config-if-range)# ipv6 snooping attach-policy <i>example_policy</i> or Switch(config-if-range)# ipv6 snooping attach-policy <i>example_policy</i> vlan 222,223,224 or Switch(config-if-range)# ipv6 snooping vlan 222,223,224	Attaches the IPv6 Snooping policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Switch#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Snooping Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping Policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 snooping** [**attach-policy** *policy_name*]

4. do show running-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	vlan configuration <i>vlan_list</i> Example: Switch(config)# vlan configuration 333	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 snooping [attach-policy <i>policy_name</i>] Example: Switch(config-vlan-config)# ipv6 snooping attach-policy example_policy	Attaches the IPv6 Snooping policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, security-level guard , device-role node , protocol ndp and dhcp .
Step 4	do show running-config Example: Switch#(config-if)# do show running-config	Verifies that the policy is attached to the specified VLANs without exiting the interface configuration mode.

How to Configure the IPv6 Binding Table Content

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. **[no] ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** *interface_type* *stack/module/port* *hw_address* [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**tracking**{ [**default** | **disable**] [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**enable** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] }]]
3. **[no] ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]]
4. **ipv6 neighbor binding logging**
5. **exit**
6. **show ipv6 neighbor binding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	[no] ipv6 neighbor binding [vlan vlan-id {ipv6-address interface interface_type stack/module/port hw_address [reachable-lifetimevalue [seconds default infinite] [tracking { [default disable] [reachable-lifetimevalue [seconds default infinite] [enable [reachable-lifetimevalue [seconds default infinite] [retry-interval {seconds} default [reachable-lifetimevalue [seconds default infinite] }] Example: Switch(config)# ipv6 neighbor binding	Adds a static entry to the binding table database.
Step 3	[no] ipv6 neighbor binding max-entries number [mac-limit number port-limit number [mac-limit number] vlan-limit number [[mac-limit number] [port-limit number [mac-limit number]]]] Example: Switch(config)# ipv6 neighbor binding max-entries 30000	Specifies the maximum number of entries that are allowed to be inserted in the binding table cache.
Step 4	ipv6 neighbor binding logging Example: Switch(config)# ipv6 neighbor binding logging	Enables the logging of binding table main events.
Step 5	exit Example: Switch(config)# exit	Exits global configuration mode, and places the router in privileged EXEC mode.
Step 6	show ipv6 neighbor binding Example: Switch# show ipv6 neighbor binding	Displays contents of a binding table.

How to Configure an IPv6 Neighbor Discovery Inspection Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 ND Inspection Policy:

SUMMARY STEPS

- 1. configure terminal**
- 2. [no]ipv6 nd inspection policy *policy-name***

3. **device-role** {*host* | *monitor* | *router* | *switch*}
4. **drop-unsecure**
5. **limit address-count** *value*
6. **sec-level minimum** *value*
7. **tracking** {*enable* [*reachable-lifetime* {*value* | *infinite*}] | *disable* [*stale-lifetime* {*value* | *infinite*}]}
8. **trusted-port**
9. **validate source-mac**
10. **no** {*device-role* | *drop-unsecure* | *limit address-count* | *sec-level minimum* | *tracking* | *trusted-port* | *validate source-mac*}
11. **default** {*device-role* | *drop-unsecure* | *limit address-count* | *sec-level minimum* | *tracking* | *trusted-port* | *validate source-mac*}
12. **do show ipv6 nd inspection policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	[no]ipv6 nd inspection policy <i>policy-name</i> Example: Switch(config)# ipv6 nd inspection policy example_policy	Specifies the ND inspection policy name and enters ND Inspection Policy configuration mode.
Step 3	device-role { <i>host</i> <i>monitor</i> <i>router</i> <i>switch</i> } Example: Switch(config-nd-inspection)# device-role switch	Specifies the role of the device attached to the port. The default is <i>host</i> .
Step 4	drop-unsecure Example: Switch(config-nd-inspection)# drop-unsecure	Drops messages with no or invalid options or an invalid signature.
Step 5	limit address-count <i>value</i> Example: Switch(config-nd-inspection)# limit address-count 1000	Enter 1–10,000.
Step 6	sec-level minimum <i>value</i> Example: Switch(config-nd-inspection)# limit address-count 1000	Specifies the minimum security level parameter value when Cryptographically Generated Address (CGA) options are used.
Step 7	tracking { <i>enable</i> [<i>reachable-lifetime</i> { <i>value</i> <i>infinite</i> }] <i>disable</i> [<i>stale-lifetime</i> { <i>value</i> <i>infinite</i> }]} Example:	Overrides the default tracking policy on a port.

	Command or Action	Purpose
	<code>Switch(config-nd-inspection)# tracking disable stale-lifetime infinite</code>	
Step 8	trusted-port Example: <code>Switch(config-nd-inspection)# trusted-port</code>	Configures a port to become a trusted port.
Step 9	validate source-mac Example: <code>Switch(config-nd-inspection)# validate source-mac</code>	Checks the source media access control (MAC) address against the link-layer address.
Step 10	no {device-role drop-unsecure limit address-count sec-level minimum tracking trusted-port validate source-mac} Example: <code>Switch(config-nd-inspection)# no validate source-mac</code>	Remove the current configuration of a parameter with the no form of the command.
Step 11	default {device-role drop-unsecure limit address-count sec-level minimum tracking trusted-port validate source-mac} Example: <code>Switch(config-nd-inspection)# default limit address-count</code>	Restores configuration to the default values.
Step 12	do show ipv6 nd inspection policy <i>policy_name</i> Example: <code>Switch(config-nd-inspection)# do show ipv6 nd inspection policy example_policy</code>	Verifies the ND Inspection Configuration without exiting ND inspection configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface

Starting with Cisco IOS XE Amsterdam 17.1.1 the IPv6 ND Inspection feature is deprecated and the SISF-based device tracking feature replaces it. For the corresponding replacement task, see *Attaching a Device Tracking Policy to an Interface* under the *Configuring SISF-Based Device Tracking* chapter in this document.

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to an interface or VLANs on an interface :

SUMMARY STEPS

1. **configure terminal**
2. **interface** *Interface_type stack/module/port*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all** }] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all** }]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters the global configuration mode.
Step 2	interface <i>Interface_type stack/module/port</i> Example: Switch(config)# <code>interface gigabitethernet 1/1/4</code>	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Switch(config-if)# <code>ipv6 nd inspection attach-policy example_policy</code> or Switch(config-if)# <code>ipv6 nd inspection attach-policy example_policy vlan 222,223,224</code> or Switch(config-if)# <code>ipv6 nd inspection vlan 222,223,224</code>	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config Example: Switch#(config-if)# <code>do show running-config</code>	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface

Starting with Cisco IOS XE Amsterdam 17.1.1 the IPv6 ND Inspection feature is deprecated and the SISF-based device tracking feature replaces it. For the corresponding replacement task, see *Attaching a Device Tracking Policy to an Interface* under the *Configuring SISF-Based Device Tracking* chapter in this document.

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Neighbor Discovery Inspection policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*

3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
4. **do show running-config interface** *portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Switch(config)# interface Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Switch(config-if-range)# ipv6 nd inspection attach-policy example_policy or Switch(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224 or Switch(config-if-range)# ipv6 nd inspection vlan 222, 223,224	Attaches the ND Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Switch#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to VLANs Globally

Starting with Cisco IOS XE Amsterdam 17.1.1 the IPv6 ND Inspection feature is deprecated and the SISF-based device tracking feature replaces it. For the corresponding replacement task, see *Attaching a Device Tracking Policy to a VLAN* under the *Configuring SISF-Based Device Tracking* chapter in this document.

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 nd inspection** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	vlan configuration <i>vlan_list</i> Example: Switch(config)# vlan configuration 334	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 nd inspection [attach-policy <i>policy_name</i>] Example: Switch(config-vlan-config)# ipv6 nd inspection attach-policy example_policy	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, device-role host , no drop-unsecure, limit address-count disabled, sec-level minimum is disabled, tracking is disabled, no trusted-port, no validate source-mac.
Step 4	do show running-config Example: Switch#(config-if)# do show running-config	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 Router Advertisement Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 Router Advertisement policy :

SUMMARY STEPS

1. **configure terminal**
2. **[no]ipv6 nd rguard policy** *policy-name*

3. [no]device-role {host | monitor | router | switch}
4. [no]hop-limit {maximum | minimum} value
5. [no]managed-config-flag {off | on}
6. [no]match {ipv6 access-list list | ra prefix-list list}
7. [no]other-config-flag {on | off}
8. [no]router-preference maximum {high | medium | low}
9. [no]trusted-port
10. default {device-role | hop-limit {maximum | minimum} | managed-config-flag | match {ipv6 access-list | ra prefix-list } | other-config-flag | router-preference maximum| trusted-port}
11. do show ipv6 nd rguard policy policy_name

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	[no]ipv6 nd rguard policy <i>policy-name</i> Example: Switch(config)# ipv6 nd rguard policy example_policy	Specifies the RA Guard policy name and enters RA Guard Policy configuration mode.
Step 3	[no]device-role {host monitor router switch} Example: Switch(config-nd-rguard) # device-role switch	Specifies the role of the device attached to the port. The default is host .
Step 4	[no]hop-limit {maximum minimum} value Example: Switch(config-nd-rguard) # hop-limit maximum 33	(1–255) Range for Maximum and Minimum Hop Limit values. Enables filtering of Router Advertisement messages by the Hop Limit value. A rogue RA message may have a low Hop Limit value (equivalent to the IPv4 Time to Live) that when accepted by the host, prevents the host from generating traffic to destinations beyond the rogue RA message generator. An RA message with an unspecified Hop Limit value is blocked. If not configured, this filter is disabled. Configure minimum to block RA messages with Hop Limit values lower than the value you specify. Configure maximum to block RA messages with Hop Limit values greater than the value you specify.
Step 5	[no]managed-config-flag {off on} Example: Switch(config-nd-rguard) # managed-config-flag on	Enables filtering of Router Advertisement messages by the Managed Address Configuration, or "M" flag field. A rogue RA message with an M field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled.

	Command or Action	Purpose
		<p>On—Accepts and forwards RA messages with an M value of 1, blocks those with 0.</p> <p>Off—Accepts and forwards RA messages with an M value of 0, blocks those with 1.</p>
Step 6	<p><code>[no]match {ipv6 access-list list ra prefix-list list}</code></p> <p>Example:</p> <pre>Switch(config-nd-raguard)# match ipv6 access-list example_list</pre>	Matches a specified prefix list or access list.
Step 7	<p><code>[no]other-config-flag {on off}</code></p> <p>Example:</p> <pre>Switch(config-nd-raguard)# other-config-flag on</pre>	<p>Enables filtering of Router Advertisement messages by the Other Configuration, or "O" flag field. A rogue RA message with an O field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled.</p> <p>On—Accepts and forwards RA messages with an O value of 1, blocks those with 0.</p> <p>Off—Accepts and forwards RA messages with an O value of 0, blocks those with 1.</p>
Step 8	<p><code>[no]router-preference maximum {high medium low}</code></p> <p>Example:</p> <pre>Switch(config-nd-raguard)# router-preference maximum high</pre>	<p>Enables filtering of Router Advertisement messages by the Router Preference flag. If not configured, this filter is disabled.</p> <ul style="list-style-type: none"> high—Accepts RA messages with the Router Preference set to high, medium, or low. medium—Blocks RA messages with the Router Preference set to high. low—Blocks RA messages with the Router Preference set to medium and high.
Step 9	<p><code>[no]trusted-port</code></p> <p>Example:</p> <pre>Switch(config-nd-raguard)# trusted-port</pre>	When configured as a trusted port, all attached devices are trusted, and no further message verification is performed.
Step 10	<p><code>default {device-role hop-limit {maximum minimum} managed-config-flag match {ipv6 access-list ra prefix-list } other-config-flag router-preference maximum trusted-port}</code></p> <p>Example:</p> <pre>Switch(config-nd-raguard)# default hop-limit</pre>	Restores a command to its default value.
Step 11	<p><code>do show ipv6 nd raguard policy policy_name</code></p> <p>Example:</p> <pre>Switch(config-nd-raguard)# do show ipv6 nd raguard policy example_policy</pre>	(Optional)—Displays the ND Guard Policy configuration without exiting the RA Guard policy configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to an interface or to VLANs on the interface :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd rguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type <i>stack/module/port</i> Example: Switch(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 nd rguard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Switch(config-if)# ipv6 nd rguard attach-policy example_policy or Switch(config-if)# ipv6 nd rguard attach-policy example_policy vlan 222,223,224 or Switch(config-if)# ipv6 nd rguard vlan 222,223,224	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config Example: Switch#(config-if)# do show running-config	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement Guard Policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd rguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
4. **do show running-config interface** *portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Switch(config)# interface Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 nd rguard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Switch(config-if-range)# ipv6 nd rguard attach-policy example_policy or Switch(config-if-range)# ipv6 nd rguard attach-policy example_policy vlan 222,223,224 or Switch(config-if-range)# ipv6 nd rguard vlan 222,223,224	Attaches the RA Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Switch#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to VLANs regardless of interface:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	vlan configuration <i>vlan_list</i> Example: Switch(config)# vlan configuration 335	Specifies the VLANs to which the IPv6 RA Guard policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy <i>policy_name</i>] Example: Switch(config-vlan-config)# ipv6 nd rguard attach-policy example_policy	Attaches the IPv6 RA Guard policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config Example: Switch#(config-if)# do show running-config	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 DHCP Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 DHCP (DHCPv6) Guard policy:

SUMMARY STEPS

1. **configure terminal**
2. **[no]ipv6 dhcp guard policy *policy-name***
3. **[no]device-role {client | server}**
4. **[no] match server access-list *ipv6-access-list-name***
5. **[no] match reply prefix-list *ipv6-prefix-list-name***
6. **[no]preference { max *limit* | min *limit* }**
7. **[no] trusted-port**
8. **default {device-role | trusted-port}**
9. **do show ipv6 dhcp guard policy *policy_name***

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	[no]ipv6 dhcp guard policy <i>policy-name</i> Example: Switch(config)# ipv6 dhcp guard policy example_policy	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.
Step 3	[no]device-role {client server} Example: Switch(config-dhcp-guard)# device-role server	(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is client . <ul style="list-style-type: none"> • client—Default value, specifies that the attached device is a client. Server messages are dropped on this port. • server—Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port.
Step 4	[no] match server access-list <i>ipv6-access-list-name</i> Example: <pre>;;Assume a preconfigured IPv6 Access List as follows: Switch(config)# ipv6 access-list my_acls Switch(config-ipv6-acl)# permit host FE80::A8BB:CCEFF:FE01:F700 any ;;configure DHCPv6 Guard to match approved access list. Switch(config-dhcp-guard)# match server access-list my_acls</pre>	(Optional). Enables verification that the advertised DHCPv6 server or relay address is from an authorized server access list (The destination address in the access list is 'any'). If not configured, this check will be bypassed. An empty access list is treated as a permit all.

	Command or Action	Purpose
Step 5	<p>[no] match reply prefix-list <i>ipv6-prefix-list-name</i></p> <p>Example:</p> <pre>;;Assume a preconfigured IPv6 prefix list as follows: Switch(config)# ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128 ;; Configure DHCPv6 Guard to match prefix Switch(config-dhcp-guard)# match reply prefix-list my_prefix</pre>	<p>(Optional) Enables verification of the advertised prefixes in DHCPv6 reply messages from the configured authorized prefix list. If not configured, this check will be bypassed. An empty prefix list is treated as a permit.</p>
Step 6	<p>[no]preference { <i>max limit</i> <i>min limit</i> }</p> <p>Example:</p> <pre>Switch(config-dhcp-guard)# preference max 250 Switch(config-dhcp-guard)#preference min 150</pre>	<p>Configure max and min when device-role is server to filter DHCPv6 server advertisements by the server preference value. The defaults permit all advertisements.</p> <p>max limit—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is less than the specified limit. Default is 255. If not specified, this check will be bypassed.</p> <p>min limit—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is greater than the specified limit. Default is 0. If not specified, this check will be bypassed.</p>
Step 7	<p>[no] trusted-port</p> <p>Example:</p> <pre>Switch(config-dhcp-guard)# trusted-port</pre>	<p>(Optional) trusted-port—Sets the port to a trusted mode. No further policing takes place on the port.</p> <p>Note If you configure a trusted port then the device-role option is not available.</p>
Step 8	<p>default {<i>device-role</i> <i>trusted-port</i>}</p> <p>Example:</p> <pre>Switch(config-dhcp-guard)# default device-role</pre>	<p>(Optional) default—Sets a command to its defaults.</p>
Step 9	<p>do show ipv6 dhcp guard policy <i>policy_name</i></p> <p>Example:</p> <pre>Switch(config-dhcp-guard)# do show ipv6 dhcp guard policy example_policy</pre>	<p>(Optional) Displays the configuration of the IPv6 DHCP guard policy without leaving the configuration submode. Omitting the <i>policy_name</i> variable displays all DHCPv6 policies.</p>

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy poll
 device-role server
 match server access-list acl1
 match reply prefix-list abc
```

```

preference min 0
preference max 255
trusted-port
interface GigabitEthernet 0/2/0
switchport
ipv6 dhcp guard attach-policy poll1 vlan add 1
vlan 1
    ipv6 dhcp guard attach-policy poll1
show ipv6 dhcp guard policy poll1

```

How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type stack/module/port
3. **ipv6 dhcp guard** [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]]
4. **do show running-config interface** Interface_type stack/module/port

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port Example: Switch(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids exceptvlan_ids none remove vlan_ids all}]] Example: Switch(config-if)# ipv6 dhcp guard attach-policy example_policy or Switch(config-if)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 or	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
	Switch(config-if)# ipv6 dhcp guard vlan 222, 223,224	
Step 4	<p>do show running-config interface Interface_type stack/module/port</p> <p>Example:</p> <pre>Switch#(config-if)# do show running-config gig 1/1/4</pre>	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** Interface_name
3. **ipv6 dhcp guard** [attach-policy policy_name [**vlan** {vlan_ids | **add** vlan_ids | **except** vlan_ids | **none** | **remove** vlan_ids | **all**}] | **vlan** [{vlan_ids | **add** vlan_ids | **except** vlan_ids | **none** | **remove** vlan_ids | **all**}]
4. **do show running-config interface** portchannel_interface_name

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters the global configuration mode.
Step 2	<p>interface range Interface_name</p> <p>Example:</p> <pre>Switch(config)# interface Po11</pre>	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	<p>ipv6 dhcp guard [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}]</p> <p>Example:</p> <pre>Switch(config-if-range)# ipv6 dhcp guard attach-policy example_policy</pre>	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
	<pre>or Switch(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 or Switch(config-if-range)#ipv6 dhcp guard vlan 222, 223,224</pre>	
Step 4	<p>do show running-config interface<i>portchannel_interface_name</i></p> <p>Example:</p> <pre>Switch#(config-if-range)# do show running-config int poll</pre>	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters the global configuration mode.
Step 2	<p>vlan configuration <i>vlan_list</i></p> <p>Example:</p> <pre>Switch(config)# vlan configuration 334</pre>	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	<p>ipv6 dhcp guard [attach-policy <i>policy_name</i>]</p> <p>Example:</p> <pre>Switch(config-vlan-config)#ipv6 dhcp guard attach-policy example_policy</pre>	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, device-role client , no trusted-port.

	Command or Action	Purpose
Step 4	do show running-config Example: Switch#(config-if)# do show running-config	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure IPv6 Source Guard

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **[no] ipv6 source-guard policy *policy_name***
4. **[deny global-autoconf] [permit link-local] [default{. . .}] [exit] [no{. . .}]**
5. **end**
6. **show ipv6 source-guard policy *policy_name***

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	[no] ipv6 source-guard policy <i>policy_name</i> Example: Device(config)# ipv6 source-guard policy example_policy	Specifies the IPv6 Source Guard policy name and enters IPv6 Source Guard policy configuration mode.
Step 4	[deny global-autoconf] [permit link-local] [default{. . .}] [exit] [no{. . .}] Example: Device(config-sisf-sourceguard)# deny global-autoconf	(Optional) Defines the IPv6 Source Guard policy. <ul style="list-style-type: none"> • deny global-autoconf—Denies data traffic from auto-configured global addresses. This is useful when all global addresses on a link are DHCP-assigned and the administrator wants to block hosts with self-configured addresses to send traffic. • permit link-local—Allows all data traffic that is sourced by a link-local address. <p>Note Trusted option under source guard policy is not supported.</p>

	Command or Action	Purpose
Step 5	end Example: Device(config-sisf-sourceguard) # end	Exits out of IPv6 Source Guard policy configuration mode.
Step 6	show ipv6 source-guard policy <i>policy_name</i> Example: Device# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

What to do next

Apply the IPv6 Source Guard policy to an interface.

How to Attach an IPv6 Source Guard Policy to an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *Interface_type stack/module/port*
4. **ipv6 source-guard** [**attach-policy** *<policy_name>*]
5. **show ipv6 source-guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface <i>Interface_type stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.

	Command or Action	Purpose
Step 4	ipv6 source-guard [attach-policy < <i>policy_name</i> >] Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device#(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to Configure IPv6 Source Guard

SUMMARY STEPS

1. enable
2. configure terminal
3. [no] ipv6 source-guard policy *policy_name*
4. [deny global-autoconf] [permit link-local] [default{. . .}] [exit] [no{. . .}]
5. end
6. show ipv6 source-guard policy *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	[no] ipv6 source-guard policy <i>policy_name</i> Example: Device(config)# ipv6 source-guard policy example_policy	Specifies the IPv6 Source Guard policy name and enters IPv6 Source Guard policy configuration mode.
Step 4	[deny global-autoconf] [permit link-local] [default{. . .}] [exit] [no{. . .}] Example:	(Optional) Defines the IPv6 Source Guard policy. <ul style="list-style-type: none"> • deny global-autoconf—Denies data traffic from auto-configured global addresses. This is useful when all global addresses on a link are DHCP-assigned and

	Command or Action	Purpose
	<pre>Device(config-sisf-sourceguard) # deny global-autoconf</pre>	<p>the administrator wants to block hosts with self-configured addresses to send traffic.</p> <ul style="list-style-type: none"> • permit link-local—Allows all data traffic that is sourced by a link-local address. <p>Note Trusted option under source guard policy is not supported.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-sisf-sourceguard) # end</pre>	Exits out of IPv6 Source Guard policy configuration mode.
Step 6	<p>show ipv6 source-guard policy <i>policy_name</i></p> <p>Example:</p> <pre>Device# show ipv6 source-guard policy example_policy</pre>	Shows the policy configuration and all the interfaces where the policy is applied.

What to do next

Apply the IPv6 Source Guard policy to an interface.

How to Attach an IPv6 Source Guard Policy to an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *Interface_type stack/module/port*
4. **ipv6 source-guard** [**attach-policy** *<policy_name>*]
5. **show ipv6 source-guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters the global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>Interface_type stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 4	ipv6 source-guard [attach-policy <policy_name>] Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name Example: Device#(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **ipv6 source-guard [attach-policy <policy_name>]**
5. **show ipv6 source-guard policy policy_name**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface port-channel <i>port-channel-number</i> Example: Device (config)# interface Po4	Specifies an interface type and port number and places the switch in the port channel configuration mode.

	Command or Action	Purpose
Step 4	ipv6 source-guard [attach-policy < <i>policy_name</i> >] Example: Device(config-if) # ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device(config-if) # show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to Configure IPv6 Prefix Guard



Note To allow routing protocol control packets sourced by a link-local address when prefix guard is applied, enable the permit link-local command in the source-guard policy configuration mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. [**no**] **ipv6 source-guard policy** *source-guard-policy*
4. [**no**] **validate address**
5. **validate prefix**
6. **exit**
7. **show ipv6 source-guard policy** [*source-guard-policy*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>[no] ipv6 source-guard policy <i>source-guard-policy</i></p> <p>Example:</p> <pre>Device(config)# ipv6 source-guard policy my_snooping_policy</pre>	Defines an IPv6 source-guard policy name and enters switch integrated security features source-guard policy configuration mode.
Step 4	<p>[no] validate address</p> <p>Example:</p> <pre>Device(config-sisf-sourceguard)# no validate address</pre>	Disables the validate address feature and enables the IPv6 prefix guard feature to be configured.
Step 5	<p>validate prefix</p> <p>Example:</p> <pre>Device(config-sisf-sourceguard)# validate prefix</pre>	Enables IPv6 source guard to perform the IPv6 prefix-guard operation.
Step 6	<p>exit</p> <p>Example:</p> <pre>Device(config-sisf-sourceguard)# exit</pre>	Exits switch integrated security features source-guard policy configuration mode and returns to privileged EXEC mode.
Step 7	<p>show ipv6 source-guard policy [<i>source-guard-policy</i>]</p> <p>Example:</p> <pre>Device# show ipv6 source-guard policy policy1</pre>	Displays the IPv6 source-guard policy configuration.

How to Attach an IPv6 Prefix Guard Policy to an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *Interface_type stack/module/port*
4. **ipv6 source-guard attach-policy** *policy_name*
5. **show ipv6 source-guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters the global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>Interface_type stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 4	ipv6 source-guard attach-policy <i>policy_name</i> Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **ipv6 source-guard** [**attach-policy** *<policy_name>*]
5. **show ipv6 source-guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface port-channel <i>port-channel-number</i> Example: Device (config)# interface Po4	Specifies an interface type and port number and places the switch in the port channel configuration mode.

	Command or Action	Purpose
Step 4	ipv6 source-guard [attach-policy <policy_name>] Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name Example: Device(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.



CHAPTER 74

Configuring FIPS

- [Information About FIPS and Common Criteria, on page 1551](#)

Information About FIPS and Common Criteria

The Federal Information Processing Standard (FIPS) certification documents for Cisco Catalyst series switches are posted on the following website:

http://www.cisco.com/web/strategy/government/security_certification/net_business_benefit_seccert_fips140.html

Click the link in the Certification column to view the Consolidated Validation Certificate and the Security Policy document. The Security Policy document describes the FIPS implementation, hardware installation, firmware initialization, and software configuration procedures for FIPS operation.

Common Criteria is an international standard (ISO/IEC 15408) for computer security certification. This standard is a set of requirements, tests, and evaluation methods that ensures that the Target of Evaluation complies with a specific Protection Profile or custom Security Target. For more information, see the security target document for specific models and IOS Releases at:

http://www.niap-ccevs.org/CCEVS_Products/pcl.cfm?tech_name=Network+Switch



CHAPTER 75

Configuring Control Plane Policing

- [Restrictions for Control Plane Policing, on page 1553](#)
- [Control Plane Policing, on page 1553](#)
- [Configuring Control Plane Policing, on page 1554](#)
- [Examples: Configuring CoPP, on page 1555](#)

Restrictions for Control Plane Policing

The following restrictions apply while Configuring Control Plane Policing:

- Only six among the following protocols can be configured simultaneously: **rip**, **ospf-v6**, **eigrp-v6**, **rip-v6**, **dhcp-snoop-client-to-server**, **dhcp-snoop-server-to-client**, **ndp-router-solicitation**, **ndp-router-advertisement**, **ndp-redirect**, **dhcpv6-client-to-server**, **dhcpv6-server-to-client**, **igrp**.
- For **ospf**, **eigrp** and **ripv2** protocols, control packets which are destined to multicast Mac of the router are policed along with the "**reserve-multicast-group**" option.

Control Plane Policing

Configure the Control Plane Policing (CoPP) feature on a predefined set of protocols to control the flow of traffic coming to the CPU. The CoPP allows you to set a rate limit on specific protocol packets. These packets are policed, and the packets that conform to the defined rate limit are permitted into the CPU. CoPP protects the packets from being routed to the CPU at an undesired rate that might impact the performance of a switch and the network. In addition, the CoPP protects the CPU from denial of service (DoS) attacks and ensures routing stability, reachability, and packet delivery. You can use Multi-Layer Switching QoS CLI to set the rate limit and policing parameters on a specific protocol.



Note CoPP is supported only on LAN BASE, IP Lite, and IP Service licenses.

Configuring Control Plane Policing

Configure the Control Plane Policing (CoPP) feature on a predefined set of protocols to control the flow of traffic coming into the CPU.

SUMMARY STEPS

1. enable
2. configure terminal
3. mls qos copp protocol { autorp-announce | autorp-discovery | bgp | cdp | cgmp | dai | dhcp-snoop-client-to-server | dhcp-snoop-server-to-client | dhcpv6-client-to-server | dhcpv6-server-to-client | eigrp | eigrp-v6 | energy-wise | igmp-gs-query | igmp-leave | igmp-query | igmp-report | igmp | ipv6-pimv2 | lldp | mld-gs-query | mld-leave | mld-query | mld-report | ndp-redirect | ndp-router-advertisement | ndp-router-solicitation | ospf | ospf-v6 | pimv1 | pxe | rep-hfl | reserve-multicast-group | rip | rip-v6 | rsvp-snoop | stp } police {pps | bps} police rate
4. end
5. show mls qos copp protocols
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	mls qos copp protocol { autorp-announce autorp-discovery bgp cdp cgmp dai dhcp-snoop-client-to-server dhcp-snoop-server-to-client dhcpv6-client-to-server dhcpv6-server-to-client eigrp eigrp-v6 energy-wise igmp-gs-query igmp-leave igmp-query igmp-report igmp ipv6-pimv2 lldp mld-gs-query mld-leave mld-query mld-report ndp-redirect ndp-router-advertisement ndp-router-solicitation ospf ospf-v6 pimv1 pxe rep-hfl reserve-multicast-group rip rip-v6 rsvp-snoop stp } police {pps bps} police rate Example: Switch (config)# mls qos copp protocol cdp police	Configures a packet policer for the specified protocol. For more details about the various parameters, please refer <i>Consolidated Platform Command Reference, Cisco IOS Release 15.2(4)E</i> .

	Command or Action	Purpose
	<pre> bps 10000 Switch(config)# mls qos copp protocol cdp police pps 500 </pre>	
Step 4	<p>end</p> <p>Example:</p> <pre> Switch(config)# end </pre>	Returns to privileged EXEC mode.
Step 5	<p>show mls qos copp protocols</p> <p>Example:</p> <pre> Switch# show mls qos copp protocols </pre>	Displays the CoPP parameters and counters for all the configured protocol.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre> Switch# copy running-config startup-config </pre>	(Optional) Saves your entries in the configuration file.

What to do next

To clear the CoPP statistics, use the **clear copp counters** command.

Examples: Configuring CoPP

The following example shows how to enable Control Plane Policing (CoPP) for a specific protocol:

```

Switch (config)# mls qos copp protocol cdp police bps ?
<8000-2000000000> Bits per second (postfix k, m, g optional; decimal point allowed)
Switch (config)# mls qos copp protocol cdp police bps 10000
Switch(config)# mls qos copp protocol cdp police pps ?
<100-100000> Packet per second
Switch(config)# mls qos copp protocol cdp police pps 500

```

The following example shows the CoPP parameters and counters for all the configured protocol:

```

Switch# show running-config | inc copp
Switch#show running-config | inc copp
mls qos copp protocol rep-hfl police pps 5600
mls qos copp protocol lldp police bps 908900
mls qos copp protocol cdp police pps 3434

```

/* Copp detailed output */

```

Switch#show mls qos copp protocols

```

```

-----
Protocol                Mode          PolicerRate      PolicerBurst
InProfilePackets      OutProfilePackets  InProfileBytes    OutProfileBytes
-----
rep-hfl                 pps           5600              5600
0                       0              0                  0

```

lldp		bps	908900	908900
0	0		0	0
cdp		pps	3434	3434
45172	0		2891008	0



PART **X**

System Management

- [Administering the System, on page 1559](#)
- [Performing Device Setup Configuration, on page 1591](#)
- [Configuring RTU Licenses, on page 1617](#)
- [Clustering Switches, on page 1625](#)
- [Configuring AVC with DNS-AS, on page 1639](#)
- [Configuring SDM Templates, on page 1663](#)
- [Configuring System Message Logs, on page 1669](#)
- [Configuring Online Diagnostics, on page 1681](#)
- [Data Sanitization, on page 1691](#)
- [Troubleshooting the Software Configuration, on page 1695](#)
- [Information About Licensing, on page 1723](#)



CHAPTER 76

Administering the System

- [Information About Administering the Device, on page 1559](#)
- [How to Administer the Device, on page 1566](#)
- [Monitoring and Maintaining Administration of the Device, on page 1586](#)
- [Configuration Examples for Device Administration, on page 1587](#)

Information About Administering the Device

System Time and Date Management

You can manage the system time and date on your device using automatic configuration methods (RTC and NTP), or manual configuration methods.



Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference* on Cisco.com.

System Clock

The basis of the time service is the system clock. This clock runs from the moment the system starts up and keeps track of the date and time.

The system clock can then be set from these sources:

- RTC
- NTP
- Manual configuration

The system clock can provide time to these services:

- User **show** commands
- Logging and debugging messages

The system clock keeps track of time internally based on Coordinated Universal Time (UTC), also known as Greenwich Mean Time (GMT). You can configure information about the local time zone and summer time (daylight saving time) so that the time appears correctly for the local time zone.

The system clock keeps track of whether the time is *authoritative* or not (that is, whether it has been set by a time source considered to be authoritative). If it is not authoritative, the time is available only for display purposes and is not redistributed.

Real Time Clock

A real-time clock (RTC) keeps track of the current time on the switch. The switch is shipped to you with RTC set to GMT time until you reconfigure clocking parameters.

The benefits of an RTC are:

- RTC is battery-powered.
- System time is retained during power outage and at system reboot.

The RTC and NTP clocks are integrated on the switch. When NTP is enabled, the RTC time is periodically synchronized to the NTP clock to maintain accuracy.

Network Time Protocol

The NTP is designed to time-synchronize a network of devices. NTP runs over User Datagram Protocol (UDP), which runs over IP. NTP is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two devices to within a millisecond of one another.

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

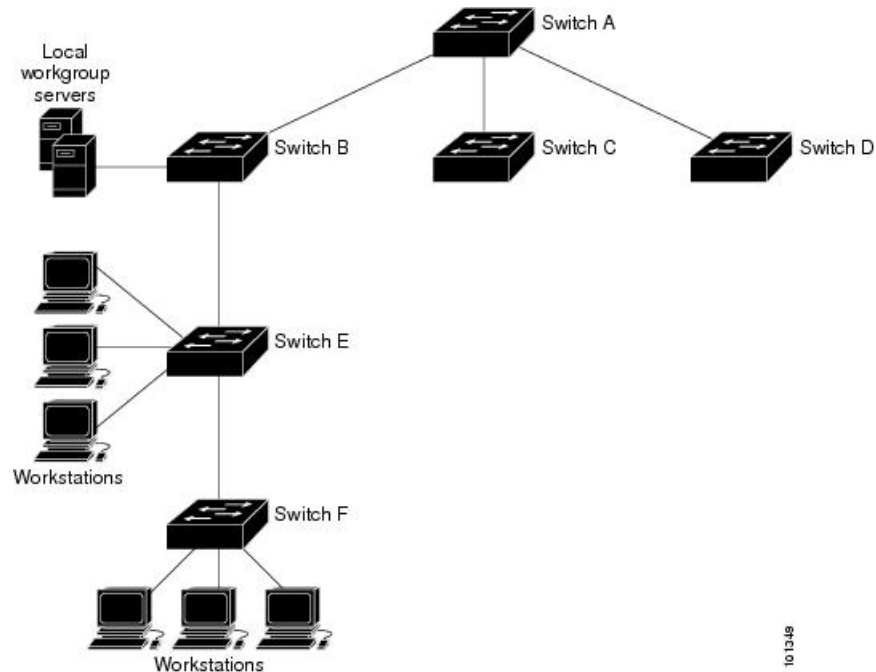
The communications between devices running NTP (known as associations) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

Cisco's implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that the time service for your network be derived from the public NTP servers available on the IP Internet.

The figure below shows a typical network example using NTP. Device A is the NTP primary (formerly known as NTP primary), with the **Device B, C, and D** configured in NTP server mode, in server association with Device A. Device E is configured as an NTP peer to the upstream and downstream Device, Device B and Device F, respectively.

Figure 120: Typical NTP Network Configuration



If the network is isolated from the Internet, Cisco's implementation of NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

NTP Stratum

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

NTP Associations

The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

NTP Security

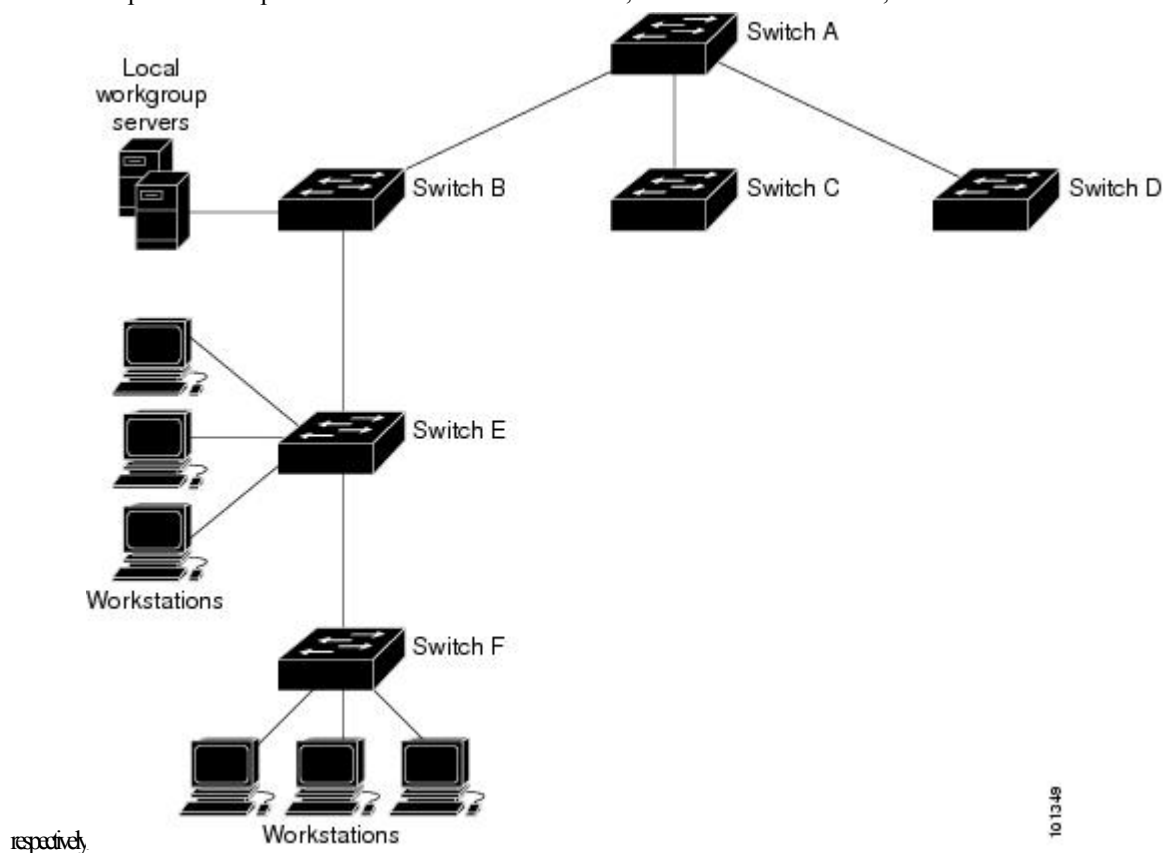
The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

NTP Implementation

Implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that the time service for your network be derived from the public NTP servers available on the IP Internet.

Figure 121: Typical NTP Network Configuration

The following figure shows a typical network example using NTP. Switch A is the NTP primary, with the Switch B, C, and D configured in NTP server mode, in server association with Switch A. Switch E is configured as an NTP peer to the upstream and downstream switches, Switch B and Switch F,



If the network is isolated from the Internet, NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

NTP Version 4

NTP version 4 is implemented on the device. NTPv4 is an extension of NTP version 3. NTPv4 supports both IPv4 and IPv6 and is backward-compatible with NTPv3.

NTPv4 provides these capabilities:

- Support for IPv6.
- Improved security compared to NTPv3. The NTPv4 protocol provides a security framework based on public key cryptography and standard X509 certificates.
- Automatic calculation of the time-distribution hierarchy for a network. Using specific multicast groups, NTPv4 automatically configures the hierarchy of the servers to achieve the best time accuracy for the lowest bandwidth cost. This feature leverages site-local IPv6 multicast addresses.

For details about configuring NTPv4, see the *Implementing NTPv4 in IPv6* chapter of the *Cisco IOS IPv6 Configuration Guide, Release 12.4T*.

System Name and Prompt

You configure the system name on the Device to identify it. By default, the system name and prompt are *Switch*.

If you have not configured a system prompt, the first 20 characters of the system name are used as the system prompt. A greater-than symbol [`>`] is appended. The prompt is updated whenever the system name changes.

For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.4* and the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

Default System Name and Prompt Configuration

The default switch system name and prompt is *Switch*.

DNS

The DNS protocol controls the Domain Name System (DNS), a distributed database with which you can map hostnames to IP addresses. When you configure DNS on your device, you can substitute the hostname for the IP address with all IP commands, such as **ping**, **telnet**, **connect**, and related Telnet support operations.

IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco Systems is a

commercial organization that IP identifies by a *com* domain name, so its domain name is *cisco.com*. A specific device in this domain, for example, the File Transfer Protocol (FTP) system is identified as *ftp.cisco.com*.

To keep track of domain names, IP has defined the concept of a domain name server, which holds a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the hostnames, specify the name server that is present on your network, and enable the DNS.

Default DNS Settings

Table 155: Default DNS Settings

Feature	Default Setting
DNS enable state	Enabled.
DNS default domain name	None configured.
DNS servers	No name server addresses are configured.

Login Banners

You can configure a message-of-the-day (MOTD) and a login banner. The MOTD banner is displayed on all connected terminals at login and is useful for sending messages that affect all network users (such as impending system shutdowns).

The login banner is also displayed on all connected terminals. It appears after the MOTD banner and before the login prompts.



Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.4*.

Default Banner Configuration

The MOTD and login banners are not configured.

MAC Address Table

The MAC address table contains address information that the device uses to forward traffic between ports. All MAC addresses in the address table are associated with one or more ports. The address table includes these types of addresses:

- Dynamic address—A source MAC address that the device learns and then ages when it is not in use.
- Static address—A manually entered unicast address that does not age and that is not lost when the device resets.

The address table lists the destination MAC address, the associated VLAN ID, and port number associated with the address and the type (static or dynamic).



Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

MAC Address Table Creation

With multiple MAC addresses supported on all ports, you can connect any port on the device to other network devices. The device provides dynamic addressing by learning the source address of packets it receives on each port and adding the address and its associated port number to the address table. As devices are added or removed from the network, the device updates the address table, adding new dynamic addresses and aging out those that are not in use.

The aging interval is globally configured. However, the device maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The device sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the device forwards the packet only to the port associated with the destination address. If the destination address is on the port that sent the packet, the packet is filtered and not forwarded. The device always uses the store-and-forward method: complete packets are stored and checked for errors before transmission.

MAC Addresses and VLANs

All addresses are associated with a VLAN. An address can exist in more than one VLAN and have different destinations in each. Unicast addresses, for example, could be forwarded to port 1 in VLAN 1 and ports 9, 10, and 1 in VLAN 5.

Each VLAN maintains its own logical address table. A known address in one VLAN is unknown in another until it is learned or statically associated with a port in the other VLAN.

Default MAC Address Table Settings

The following table shows the default settings for the MAC address table.

Table 156: Default Settings for the MAC Address

Feature	Default Setting
Aging time	300 seconds
Dynamic addresses	Automatically learned
Static addresses	None configured

ARP Table Management

To communicate with a device (over Ethernet, for example), the software first must learn the 48-bit MAC address or the local data link address of that device. The process of learning the local data link address from an IP address is called *address resolution*.

The Address Resolution Protocol (ARP) associates a host IP address with the corresponding media or MAC addresses and the VLAN ID. Using an IP address, ARP finds the associated MAC address. When a MAC

address is found, the IP-MAC address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests and replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP). By default, standard Ethernet-style ARP encapsulation (represented by the **arpa** keyword) is enabled on the IP interface.

ARP entries added manually to the table do not age and must be manually removed.

For CLI procedures, see the Cisco IOS Release 12.4 documentation on *Cisco.com*.

How to Administer the Device

Configuring the Time and Date Manually

System time remains accurate through restarts and reboot, however, you can manually configure the time and date after the system is restarted.

We recommend that you use manual configuration only when necessary. If you have an outside source to which the device can synchronize, you do not need to manually set the system clock.

Setting the System Clock

If you have an outside source on the network that provides time services, such as an NTP server, you do not need to manually set the system clock.

Follow these steps to set the system clock:

SUMMARY STEPS

1. **enable**
2. Use one of the following:
 - **clock set** *hh:mm:ss day month year*
 - **clock set** *hh:mm:ss month day year*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	Use one of the following: <ul style="list-style-type: none"> • clock set <i>hh:mm:ss day month year</i> • clock set <i>hh:mm:ss month day year</i> Example:	Manually set the system clock using one of these formats: <ul style="list-style-type: none"> • <i>hh:mm:ss</i>—Specifies the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone. • <i>day</i>—Specifies the day by date in the month.

	Command or Action	Purpose
	Switch# <code>clock set 13:32:00 23 March 2013</code>	<ul style="list-style-type: none"> • <i>month</i>—Specifies the month by name. • <i>year</i>—Specifies the year (no abbreviation).

Configuring the Time Zone

Follow these steps to manually configure the time zone:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `clock timezone zone hours-offset [minutes-offset]`
4. `end`
5. `show running-config`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	clock timezone zone hours-offset [minutes-offset] Example: Switch(config)# <code>clock timezone AST -3 30</code>	Sets the time zone. Internal time is kept in Coordinated Universal Time (UTC), so this command is used only for display purposes and when the time is manually set. <ul style="list-style-type: none"> • <i>zone</i>—Enters the name of the time zone to be displayed when standard time is in effect. The default is UTC. • <i>hours-offset</i>—Enters the hours offset from UTC. • (Optional) <i>minutes-offset</i>—Enters the minutes offset from UTC. This available where the local time zone is a percentage of an hour different from UTC.
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Summer Time (Daylight Saving Time)

To configure summer time (daylight saving time) in areas where it starts and ends on a particular day of the week each year, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock summer-time zone date date month year hh:mm date month year hh:mm [offset]**
4. **clock summer-time zone recurring [week day month hh:mm week day month hh:mm [offset]]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	clock summer-time zone date date month year hh:mm date month year hh:mm [offset]	Configures summer time to start and end on specified days every year.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(config)# clock summer-time PDT date 10 March 2013 2:00 3 November 2013 2:00</pre>	
Step 4	<p>clock summer-time <i>zone</i> recurring [<i>week day month hh:mm week day month hh:mm [offset]</i>]</p> <p>Example:</p> <pre>Switch(config)# clock summer-time PDT recurring 10 March 2013 2:00 3 November 2013 2:00</pre>	<p>Configures summer time to start and end on the specified days every year. All times are relative to the local time zone. The start time is relative to standard time.</p> <p>The end time is relative to summer time. Summer time is disabled by default. If you specify clock summer-time zone recurring without parameters, the summer time rules default to the United States rules.</p> <p>If the starting month is after the ending month, the system assumes that you are in the southern hemisphere.</p> <ul style="list-style-type: none"> • <i>zone</i>—Specifies the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) <i>week</i>— Specifies the week of the month (1 to 4, first, or last). • (Optional) <i>day</i>—Specifies the day of the week (Sunday, Monday...). • (Optional) <i>month</i>—Specifies the month (January, February...). • (Optional) <i>hh:mm</i>—Specifies the time (24-hour format) in hours and minutes. • (Optional) <i>offset</i>—Specifies the number of minutes to add during summer time. The default is 60.
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Follow these steps if summer time in your area does not follow a recurring pattern (configure the exact date and time of the next summer time events):

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock summer-time zone date** [*month date year hh:mm month date year hh:mm [offset]*] **or** **clock summer-time zone date** [*date month year hh:mm date month year hh:mm [offset]*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	clock summer-time zone date [<i>month date year hh:mm month date year hh:mm [offset]</i>] or clock summer-time zone date [<i>date month year hh:mm date month year hh:mm [offset]</i>]	Configures summer time to start on the first date and end on the second date. Summer time is disabled by default. <ul style="list-style-type: none"> • For <i>zone</i>, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For <i>week</i>, specify the week of the month (1 to 5 or last). • (Optional) For <i>day</i>, specify the day of the week (Sunday, Monday...). • (Optional) For <i>month</i>, specify the month (January, February...).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) For <i>hh:mm</i>, specify the time (24-hour format) in hours and minutes. • (Optional) For <i>offset</i>, specify the number of minutes to add during summer time. The default is 60.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a System Name

Follow these steps to manually configure a system name:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname *name***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	hostname <i>name</i> Example: <pre>Switch(config)# hostname remote-users</pre>	<p>Configures a system name. When you set the system name, it is also used as the system prompt.</p> <p>The default setting is Switch.</p> <p>The name must follow the rules for ARPANET hostnames. They must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens. Names can be up to 63 characters.</p>
Step 4	end Example: <pre>remote-users(config)#end remote-users#</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting Up DNS

If you use the device IP address as its hostname, the IP address is used and no DNS query occurs. If you configure a hostname that contains no periods (.), a period followed by the default domain name is appended to the hostname before the DNS query is made to map the name to an IP address. The default domain name is the value set by the **ip domain-name** global configuration command. If there is a period (.) in the hostname, the Cisco IOS software looks up the IP address without appending any default domain name to the hostname.

Follow these steps to set up your switch to use the DNS:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip domain-name *name***
4. **ip name-server *server-address1* [*server-address2* ... *server-address6*]**

5. **ip domain-lookup** [nsap | source-interface *interface*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip domain-name <i>name</i> Example: <pre>Switch(config)# ip domain-name Cisco.com</pre>	Defines a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name). Do not include the initial period that separates an unqualified name from the domain name. At boot time, no domain name is configured; however, if the device configuration comes from a BOOTP or Dynamic Host Configuration Protocol (DHCP) server, then the default domain name might be set by the BOOTP or DHCP server (if the servers were configured with this information).
Step 4	ip name-server <i>server-address1</i> [<i>server-address2</i> ... <i>server-address6</i>] Example: <pre>Switch(config)# ip name-server 192.168.1.100 192.168.1.200 192.168.1.300</pre>	Specifies the address of one or more name servers to use for name and address resolution. You can specify up to six name servers. Separate each server address with a space. The first server specified is the primary server. The device sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 5	ip domain-lookup [nsap source-interface <i>interface</i>] Example: <pre>Switch(config)# ip domain-lookup</pre>	(Optional) Enables DNS-based hostname-to-address translation on your device. This feature is enabled by default. If your network devices require connectivity with devices in networks for which you do not control name assignment, you can dynamically assign device names that uniquely identify your devices by using the global Internet naming scheme (DNS).

	Command or Action	Purpose
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Switch# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Message-of-the-Day Login Banner

You can create a single or multiline message banner that appears on the screen when someone logs in to the device.

Follow these steps to configure a MOTD login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner motd *c message c***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 3	banner motd <i>c message c</i> Example: <pre>Switch(config)# banner motd # This is a secure site. Only authorized users are allowed. For access, contact technical support. #</pre>	Specifies the message of the day. <i>c</i> —Enters the delimiting character of your choice, for example, a pound sign (#), and press the Return key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. <i>message</i> —Enters a banner message up to 255 characters. You cannot use the delimiting character in the message.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner appears after the MOTD banner and before the login prompt.

Follow these steps to configure a login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner login** *c message c*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	banner login c message c Example: <pre>Switch(config)# banner login \$ Access for authorized users only. Please enter your username and password. \$</pre>	Specifies the login message. <p><i>c</i>— Enters the delimiting character of your choice, for example, a pound sign (#), and press the Return key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded.</p> <p><i>message</i>—Enters a login message up to 255 characters. You cannot use the delimiting character in the message.</p>
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Managing the MAC Address Table

Changing the Address Aging Time

Follow these steps to configure the dynamic address table aging time:

SUMMARY STEPS

1. enable
2. configure terminal
3. mac address-table aging-time [0 | 10-1000000] [routed-mac | vlan vlan-id]
4. end
5. show running-config
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>mac address-table aging-time [0 10-1000000] [routed-mac vlan vlan-id]</p> <p>Example:</p> <pre>Switch(config)# mac address-table aging-time 500 vlan 2</pre>	<p>Sets the length of time that a dynamic entry remains in the MAC address table after the entry is used or updated.</p> <p>The range is 10 to 1000000 seconds. The default is 300. You can also enter 0, which disables aging. Static address entries are never aged or removed from the table.</p> <p><i>vlan-id</i>—Valid IDs are 1 to 4094.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	<p>Verifies your entries.</p>
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Configuring MAC Address Change Notification Traps

Follow these steps to configure the switch to send MAC address change notification traps to an NMS host:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host** *host-addr* *community-string* *notification-type* { **informs** | **traps** } { **version** { **1** | **2c** | **3** } } { **vrf** *vrf instance name* }
4. **snmp-server enable traps mac-notification change**
5. **mac address-table notification change**
6. **mac address-table notification change** [*interval value*] [*history-size value*]
7. **interface** *interface-id*
8. **snmp trap mac-notification change** {**added** | **removed**}
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	snmp-server host <i>host-addr</i> <i>community-string</i> <i>notification-type</i> { informs traps } { version { 1 2c 3 } } { vrf <i>vrf instance name</i> } Example: <pre>Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	Specifies the recipient of the trap message. <ul style="list-style-type: none"> • <i>host-addr</i>—Specifies the name or address of the NMS. • traps (the default)—Sends SNMP traps to the host. • informs—Sends SNMP informs to the host. • version—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • <i>community-string</i>—Specifies the string to send with the notification operation. Though you can set this string by using the snmp-server host command, we recommend that you define this string by using the snmp-server community command before using the snmp-server host command.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>notification-type</i>—Uses the mac-notification keyword. • vrf <i>vrf instance name</i>—Specifies the VPN routing/forwarding instance for this host.
Step 4	snmp-server enable traps mac-notification change Example: <pre>Switch(config)# snmp-server enable traps mac-notification change</pre>	Enables the device to send MAC address change notification traps to the NMS.
Step 5	mac address-table notification change Example: <pre>Switch(config)# mac address-table notification change</pre>	Enables the MAC address change notification feature.
Step 6	mac address-table notification change [interval value] [history-size value] Example: <pre>Switch(config)# mac address-table notification change interval 123 Switch(config)# mac address-table notification change history-size 100</pre>	<p>Enters the trap interval time and the history table size.</p> <ul style="list-style-type: none"> • (Optional) interval value—Specifies the notification trap interval in seconds between each set of traps that are generated to the NMS. The range is 0 to 2147483647 seconds; the default is 1 second. • (Optional) history-size value—Specifies the maximum number of entries in the MAC notification history table. The range is 0 to 500; the default is 1.
Step 7	interface interface-id Example: <pre>Switch(config)# interface gigabitethernet 1/0/2</pre>	Enters interface configuration mode, and specifies the Layer 2 interface on which to enable the SNMP MAC address notification trap.
Step 8	snmp trap mac-notification change {added removed} Example: <pre>Switch(config-if)# snmp trap mac-notification change added</pre>	<p>Enables the MAC address change notification trap on the interface.</p> <ul style="list-style-type: none"> • Enables the trap when a MAC address is added on this interface. • Enables the trap when a MAC address is removed from this interface.
Step 9	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 10	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 11	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring MAC Address Move Notification Traps

When you configure MAC-move notification, an SNMP notification is generated and sent to the network management system whenever a MAC address moves from one port to another within the same VLAN.

Follow these steps to configure the device to send MAC address-move notification traps to an NMS host:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host *host-addr* {traps | informs} {version {1 | 2c | 3}} *community-string notification-type***
4. **snmp-server enable traps mac-notification move**
5. **mac address-table notification mac-move**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	snmp-server host <i>host-addr</i> {traps informs} {version {1 2c 3}} <i>community-string notification-type</i>	Specifies the recipient of the trap message. <ul style="list-style-type: none"> • <i>host-addr</i>—Specifies the name or address of the NMS.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<ul style="list-style-type: none"> • traps (the default)—Sends SNMP traps to the host. • informs—Sends SNMP informs to the host. • version—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • <i>community-string</i>—Specifies the string to send with the notification operation. Though you can set this string by using the snmp-server host command, we recommend that you define this string by using the snmp-server community command before using the snmp-server host command. • <i>notification-type</i>—Uses the mac-notification keyword.
Step 4	<p>snmp-server enable traps mac-notification move</p> <p>Example:</p> <pre>Switch(config)# snmp-server enable traps mac-notification move</pre>	Enables the device to send MAC address move notification traps to the NMS.
Step 5	<p>mac address-table notification mac-move</p> <p>Example:</p> <pre>Switch(config)# mac address-table notification mac-move</pre>	Enables the MAC address move notification feature.
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Switch# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To disable MAC address-move notification traps, use the **no snmp-server enable traps mac-notification move** global configuration command. To disable the MAC address-move notification feature, use the **no mac address-table notification mac-move** global configuration command.

You can verify your settings by entering the **show mac address-table notification mac-move** privileged EXEC commands.

Configuring MAC Threshold Notification Traps

When you configure MAC threshold notification, an SNMP notification is generated and sent to the network management system when a MAC address table threshold limit is reached or exceeded.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host *host-addr* {traps / informs} {version {1 | 2c | 3}} *community-string notification-type***
3. **snmp-server enable traps mac-notification threshold**
4. **mac address-table notification threshold**
5. **mac address-table notification threshold [*limit percentage*] | [*interval time*]**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	snmp-server host <i>host-addr</i> {traps / informs} {version {1 2c 3}} <i>community-string notification-type</i> Example: <pre>Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	Specifies the recipient of the trap message. <ul style="list-style-type: none"> • <i>host-addr</i>—Specifies the name or address of the NMS. • traps (the default)—Sends SNMP traps to the host. • informs—Sends SNMP informs to the host. • version—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • <i>community-string</i>—Specifies the string to send with the notification operation. You can set this string by using the snmp-server host command, but we recommend that you define this string by using the snmp-server community command before using the snmp-server host command. • <i>notification-type</i>—Uses the mac-notification keyword.

	Command or Action	Purpose
Step 3	snmp-server enable traps mac-notification threshold Example: <pre>Switch(config)# snmp-server enable traps mac-notification threshold</pre>	Enables MAC threshold notification traps to the NMS.
Step 4	mac address-table notification threshold Example: <pre>Switch(config)# mac address-table notification threshold</pre>	Enables the MAC address threshold notification feature.
Step 5	mac address-table notification threshold [limit percentage] [interval time] Example: <pre>Switch(config)# mac address-table notification threshold interval 123 Switch(config)# mac address-table notification threshold limit 78</pre>	Enters the threshold value for the MAC address threshold usage monitoring. <ul style="list-style-type: none"> • (Optional) limit percentage—Specifies the percentage of the MAC address table use; valid values are from 1 to 100 percent. The default is 50 percent. • (Optional) interval time—Specifies the time between notifications; valid values are greater than or equal to 120 seconds. The default is 120 seconds.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Adding and Removing Static Address Entries

Follow these steps to add a static address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table static mac-addr vlan vlan-id interface interface-id**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	mac address-table static mac-addr vlan vlan-id interface interface-id Example: Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1	Adds a static address to the MAC address table. <ul style="list-style-type: none"> • <i>mac-addr</i>—Specifies the destination MAC unicast address to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface. • <i>vlan-id</i>—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094. • <i>interface-id</i>—Specifies the interface to which the received packet is forwarded. Valid interfaces include physical ports or port channels. For static multicast addresses, you can enter multiple interface IDs. For static unicast addresses, you can enter only one interface at a time, but you can enter the command multiple times with the same MAC address and VLAN ID.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 5	show running-config Example: Switch# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Unicast MAC Address Filtering

Follow these steps to configure the Device to drop a source or destination unicast static address:

SUMMARY STEPS

1. enable
2. configure terminal
3. mac address-table static *mac-addr* vlan *vlan-id* drop
4. end
5. show running-config
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	mac address-table static <i>mac-addr</i> vlan <i>vlan-id</i> drop Example: <pre>Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop</pre>	Enables unicast MAC address filtering and configure the device to drop a packet with the specified source or destination unicast static address. <ul style="list-style-type: none"> • <i>mac-addr</i>—Specifies a source or destination unicast MAC address (48-bit). Packets with this MAC address are dropped. • <i>vlan-id</i>—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# <code>copy running-config startup-config</code>	

Monitoring and Maintaining Administration of the Device

Command	Purpose
<code>clear mac address-table dynamic</code>	Removes all dynamic entries.
<code>clear mac address-table dynamic address <i>mac-address</i></code>	Removes a specific MAC address.
<code>clear mac address-table dynamic interface <i>interface-id</i></code>	Removes all addresses on the specified physical port or port channel.
<code>clear mac address-table dynamic vlan <i>vlan-id</i></code>	Removes all addresses on a specified VLAN.
<code>show clock [<i>detail</i>]</code>	Displays the time and date configuration.
<code>show ip igmp snooping groups</code>	Displays the Layer 2 multicast entries for all VLANs or the specified VLAN.
<code>show mac address-table address <i>mac-address</i></code>	Displays MAC address table information for the specified MAC address.
<code>show mac address-table aging-time</code>	Displays the aging time in all VLANs or the specified VLAN.
<code>show mac address-table count</code>	Displays the number of addresses present in all VLANs or the specified VLAN.
<code>show mac address-table dynamic</code>	Displays only dynamic MAC address table entries.
<code>show mac address-table interface <i>interface-name</i></code>	Displays the MAC address table information for the specified interface.
<code>show mac address-table move update</code>	Displays the MAC address table move update information.
<code>show mac address-table multicast</code>	Displays a list of multicast MAC addresses.
<code>show mac address-table notification {change mac-move threshold}</code>	Displays the MAC notification parameters and history table.
<code>show mac address-table secure</code>	Displays the secure MAC addresses.
<code>show mac address-table static</code>	Displays only static MAC address table entries.
<code>show mac address-table vlan <i>vlan-id</i></code>	Displays the MAC address table information for the specified VLAN.

Configuration Examples for Device Administration

Example: Setting the System Clock

This example shows how to manually set the system clock:

```
Switch# clock set 13:32:00 23 July 2013
```

Examples: Configuring Summer Time

This example (for daylight savings time) shows how to specify that summer time starts on March 10 at 02:00 and ends on November 3 at 02:00:

```
Switch(config)# clock summer-time PDT recurring PST date  
10 March 2013 2:00 3 November 2013 2:00
```

This example shows how to set summer time start and end dates:

```
Switch(config)#clock summer-time PST date  
20 March 2013 2:00 20 November 2013 2:00
```

Example: Configuring a MOTD Banner

This example shows how to configure a MOTD banner by using the pound sign (#) symbol as the beginning and ending delimiter:

```
Switch(config)# banner motd #  
  
This is a secure site. Only authorized users are allowed.  
For access, contact technical support.  
  
#  
  
Switch(config)#
```

This example shows the banner that appears from the previous configuration:

```
Unix> telnet 192.0.2.15  
  
Trying 192.0.2.15...  
Connected to 192.0.2.15.  
Escape character is '^]'.  
  
This is a secure site. Only authorized users are allowed.  
For access, contact technical support.
```

```
User Access Verification

Password:
```

Example: Configuring a Login Banner

This example shows how to configure a login banner by using the dollar sign (\$) symbol as the beginning and ending delimiter:

```
Switch(config)# banner login $

Access for authorized users only. Please enter your username and password.

$

Switch(config)#
```

Example: Configuring MAC Address Change Notification Traps

This example shows how to specify 172.20.10.10 as the NMS, enable MAC address notification traps to the NMS, enable the MAC address-change notification feature, set the interval time to 123 seconds, set the history-size to 100 entries, and enable traps whenever a MAC address is added on the specified port:

```
Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification
Switch(config)# snmp-server enable traps mac-notification change
Switch(config)# mac address-table notification change
Switch(config)# mac address-table notification change interval 123
Switch(config)# mac address-table notification change history-size 100
Switch(config)# interface gigabitethernet 1/2/1
Switch(config-if)# snmp trap mac-notification change added
```

Example: Configuring MAC Threshold Notification Traps

This example shows how to specify 172.20.10.10 as the NMS, enable the MAC address threshold notification feature, set the interval time to 123 seconds, and set the limit to 78 per cent:

```
Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification
Switch(config)# snmp-server enable traps mac-notification threshold
Switch(config)# mac address-table notification threshold
Switch(config)# mac address-table notification threshold interval 123
Switch(config)# mac address-table notification threshold limit 78
```

Example: Adding the Static Address to the MAC Address Table

This example shows how to add the static address c2f3.220a.12f4 to the MAC address table. When a packet is received in VLAN 4 with this MAC address as its destination address, the packet is forwarded to the specified port:



Note You cannot associate the same static MAC address to multiple interfaces. If the command is executed again with a different interface, the static MAC address is overwritten on the new interface.

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/1/1
```

Example: Configuring Unicast MAC Address Filtering

This example shows how to enable unicast MAC address filtering and how to configure drop packets that have a source or destination address of c2f3.220a.12f4. When a packet is received in VLAN 4 with this MAC address as its source or destination, the packet is dropped:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```




CHAPTER 77

Performing Device Setup Configuration

- [Information About Performing Device Setup Configuration, on page 1591](#)
- [How to Perform Device Setup Configuration, on page 1601](#)
- [Monitoring Device Setup Configuration, on page 1613](#)
- [Configuration Examples for Performing Device Setup, on page 1614](#)

Information About Performing Device Setup Configuration

Review the sections in this module before performing your initial device configuration tasks that include IP address assignments and DHCP autoconfiguration.

Boot Process

To start your device, you need to follow the procedures in the getting started guide or the hardware installation guide for installing and powering on the device and setting up the initial device configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth).

The boot loader software performs the normal boot process and includes these activities:

- Locates the bootable (base) package in the bundle or installed package set.
- Performs low-level CPU initialization. It initializes the CPU registers, which control where physical memory is mapped, its quantity, its speed, and so forth.
- Performs power-on self-test (POST) for the CPU subsystem and tests the system DRAM.
- Initializes the file systems on the system board.
- Loads a default operating system software image into memory and boots up the device.

The boot loader provides access to the flash file systems before the operating system is loaded. Normally, the boot loader is used only to load, decompress, and start the operating system. After the boot loader gives the operating system control of the CPU, the boot loader is not active until the next system reset or power-on.

The boot loader also provides trap-door access into the system if the operating system has problems serious enough that it cannot be used. The trap-door operation provides enough access to the system so that if it is necessary, you can format the flash file system, reinstall the operating system software image by using the Xmodem Protocol, recover from a lost or forgotten password, and finally restart the operating system.

Before you can assign device information, make sure that you have connected a PC or terminal to the console port or a PC to the Ethernet management port, and make sure you have configured the PC or terminal-emulation software baud rate and character format to match that of the device console port settings:

- Baud rate default is 9600.
- Data bits default is 8.



Note If the data bits option is set to 8, set the parity option to none.

- Stop bits default is 2 (minor).
- Parity settings default is none.

Devices Information Assignment

You can assign IP information through the device setup program, through a DHCP server, or manually.

Use the device setup program if you want to be prompted for specific IP information. With this program, you can also configure a hostname and an enable secret password.

It gives you the option of assigning a Telnet password (to provide security during remote management) and configuring your switch as a command or member switch of a cluster or as a standalone switch.

Use a DHCP server for centralized control and automatic assignment of IP information after the server is configured.



Note If you are using DHCP, do not respond to any of the questions in the setup program until the device receives the dynamically assigned IP address and reads the configuration file.

If you are an experienced user familiar with the device configuration steps, manually configure the device. Otherwise, use the setup program described in the *Boot Process* section.

Default Switch Information

Table 157: Default Switch Information

Feature	Default Setting
IP address and subnet mask	No IP address or subnet mask are defined.
Default gateway	No default gateway is defined.
Enable secret password	No password is defined.
Hostname	The factory-assigned default hostname is Device.
Telnet password	No password is defined.

Feature	Default Setting
Cluster command switch functionality	Disabled.
Cluster name	No cluster name is defined.

DHCP-Based Autoconfiguration Overview

DHCP provides configuration information to Internet hosts and internetworking devices. This protocol consists of two components: one for delivering configuration parameters from a DHCP server to a device and an operation for allocating network addresses to devices. DHCP is built on a client-server model, in which designated DHCP servers allocate network addresses and deliver configuration parameters to dynamically configured devices. The device can act as both a DHCP client and a DHCP server.

During DHCP-based autoconfiguration, your device (DHCP client) is automatically configured at startup with IP address information and a configuration file.

With DHCP-based autoconfiguration, no DHCP client-side configuration is needed on your device. However, you need to configure the DHCP server for various lease options associated with IP addresses.

If you want to use DHCP to relay the configuration file location on the network, you might also need to configure a Trivial File Transfer Protocol (TFTP) server and a Domain Name System (DNS) server.

The DHCP server for your device can be on the same LAN or on a different LAN than the device. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your device and the DHCP server. A relay device forwards broadcast traffic between two directly connected LANs. A router does not forward broadcast packets, but it forwards packets based on the destination IP address in the received packet.

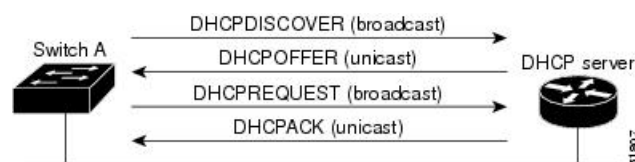
DHCP-based autoconfiguration replaces the BOOTP client functionality on your device.

DHCP Client Request Process

When you boot up your device, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the device. If the configuration file is present and the configuration includes the **ip address dhcp** interface configuration command on specific routed interfaces, the DHCP client is invoked and requests the IP address information for those interfaces.

This is the sequence of messages that are exchanged between the DHCP client and the DHCP server.

Figure 122: DHCP Client and Server Message Exchange



The client, Device A, broadcasts a DHCPDISCOVER message to locate a DHCP server. The DHCP server offers configuration parameters (such as an IP address, subnet mask, gateway IP address, DNS IP address, a lease for the IP address, and so forth) to the client in a DHCPOFFER unicast message.

In a DHCPREQUEST broadcast message, the client returns a formal request for the offered configuration information to the DHCP server. The formal request is broadcast so that all other DHCP servers that received

the DHCPDISCOVER broadcast message from the client can reclaim the IP addresses that they offered to the client.

The DHCP server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client. With this message, the client and server are bound, and the client uses configuration information received from the server. The amount of information the device receives depends on how you configure the DHCP server.

If the configuration parameters sent to the client in the DHCP OFFER unicast message are invalid (a configuration error exists), the client returns a DHCPDECLINE broadcast message to the DHCP server.

The DHCP server sends the client a DHCPNAK denial broadcast message, which means that the offered configuration parameters have not been assigned, that an error has occurred during the negotiation of the parameters, or that the client has been slow in responding to the DHCP OFFER message (the DHCP server assigned the parameters to another client).

A DHCP client might receive offers from multiple DHCP or BOOTP servers and can accept any of the offers; however, the client usually accepts the first offer it receives. The offer from the DHCP server is not a guarantee that the IP address is allocated to the client; however, the server usually reserves the address until the client has had a chance to formally request the address. If the device accepts replies from a BOOTP server and configures itself, the device broadcasts, instead of unicasts, TFTP requests to obtain the device configuration file.

The DHCP hostname option allows a group of devices to obtain hostnames and a standard configuration from the central management DHCP server. A client (device) includes in its DHCPDISCOVER message an option 12 field used to request a hostname and other configuration parameters from the DHCP server. The configuration files on all clients are identical except for their DHCP-obtained hostnames.

If a client has a default hostname (the **hostname name** global configuration command is not configured or the **no hostname** global configuration command is entered to remove the hostname), the DHCP hostname option is not included in the packet when you enter the **ip address dhcp** interface configuration command. In this case, if the client receives the DHCP hostname option from the DHCP interaction while acquiring an IP address for an interface, the client accepts the DHCP hostname option and sets the flag to show that the system now has a hostname configured.

DHCP-based Autoconfiguration and Image Update

You can use the DHCP image upgrade features to configure a DHCP server to download both a new image and a new configuration file to one or more devices in a network. Simultaneous image and configuration upgrade for all switches in the network helps ensure that each new device added to a network receives the same image and configuration.

There are two types of DHCP image upgrades: DHCP autoconfiguration and DHCP auto-image update.

Restrictions for DHCP-based Autoconfiguration

- The DHCP-based autoconfiguration with a saved configuration process stops if there is not at least one Layer 3 interface in an up state without an assigned IP address in the network.
- Unless you configure a timeout, the DHCP-based autoconfiguration with a saved configuration feature tries indefinitely to download an IP address.
- The auto-install process stops if a configuration file cannot be downloaded or if the configuration file is corrupted.

- The configuration file that is downloaded from TFTP is merged with the existing configuration in the running configuration but is not saved in the NVRAM unless you enter the **write memory** or **copy running-configuration startup-configuration** privileged EXEC command. If the downloaded configuration is saved to the startup configuration, the feature is not triggered during subsequent system restarts.

DHCP Autoconfiguration

DHCP autoconfiguration downloads a configuration file to one or more devices in your network from a DHCP server. The downloaded configuration file becomes the running configuration of the device. It does not overwrite the bootup configuration saved in the flash, until you reload the device.

DHCP Auto-Image Update

You can use DHCP auto-image upgrade with DHCP autoconfiguration to download both a configuration and a new image to one or more devices in your network. The device (or devices) downloading the new configuration and the new image can be blank (or only have a default factory configuration loaded).

If the new configuration is downloaded to a switch that already has a configuration, the downloaded configuration is appended to the configuration file stored on the switch. (Any existing configuration is not overwritten by the downloaded one.)

To enable a DHCP auto-image update on the device, the TFTP server where the image and configuration files are located must be configured with the correct option 67 (the configuration filename), option 66 (the DHCP server hostname) option 150 (the TFTP server address), and option 125 (description of the Cisco IOS image file) settings.

After you install the device in your network, the auto-image update feature starts. The downloaded configuration file is saved in the running configuration of the device, and the new image is downloaded and installed on the device. When you reboot the device, the configuration is stored in the saved configuration on the device.

DHCP Server Configuration Guidelines

Follow these guidelines if you are configuring a device as a DHCP server:

- You should configure the DHCP server with reserved leases that are bound to each device by the device hardware address.
- If you want the device to receive IP address information, you must configure the DHCP server with these lease options:
 - IP address of the client (required)
 - Subnet mask of the client (required)
 - DNS server IP address (optional)
 - Router IP address (default gateway address to be used by the device) (required)
- If you want the device to receive the configuration file from a TFTP server, you must configure the DHCP server with these lease options:
 - TFTP server name (required)
 - Boot filename (the name of the configuration file that the client needs) (recommended)

- Hostname (optional)
- Depending on the settings of the DHCP server, the device can receive IP address information, the configuration file, or both.
- If you do not configure the DHCP server with the lease options described previously, it replies to client requests with only those parameters that are configured. If the IP address and the subnet mask are not in the reply, the device is not configured. If the router IP address or the TFTP server name are not found, the device might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.
- The device can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your device but are not configured. (These features are not operational.)

Purpose of the DNS Server

The DHCP server uses the DNS server to resolve the TFTP server name to an IP address. You must configure the TFTP server name-to-IP address map on the DNS server. The TFTP server contains the configuration files for the device.

You can configure the IP addresses of the DNS servers in the lease database of the DHCP server from where the DHCP replies will retrieve them. You can enter up to two DNS server IP addresses in the lease database.

The DNS server can be on the same LAN or on a different LAN from the device. If it is on a different LAN, the device must be able to access it through a router.

How to Obtain Configuration Files

Depending on the availability of the IP address and the configuration filename in the DHCP reserved lease, the device obtains its configuration information in these ways:

- The IP address and the configuration filename is reserved for the device and provided in the DHCP reply (one-file read method).

The device receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The device sends a unicast message to the TFTP server to retrieve the named configuration file from the base directory of the server and upon receipt, it completes its boot up process.

- The IP address and the configuration filename is reserved for the device, but the TFTP server address is not provided in the DHCP reply (one-file read method).

The device receives its IP address, subnet mask, and the configuration filename from the DHCP server. The device sends a broadcast message to a TFTP server to retrieve the named configuration file from the base directory of the server, and upon receipt, it completes its boot-up process.

- Only the IP address is reserved for the device and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

The device receives its IP address, subnet mask, and the TFTP server address from the DHCP server. The device sends a unicast message to the TFTP server to retrieve the network-config or ciscoconet.cfg default configuration file. (If the network-config file cannot be read, the device reads the ciscoconet.cfg file.)

The default configuration file contains the hostnames-to-IP-address mapping for the device. The device fills its host table with the information in the file and obtains its hostname. If the hostname is not found

in the file, the device uses the hostname in the DHCP reply. If the hostname is not specified in the DHCP reply, the device uses the default *Switch* as its hostname.

After obtaining its hostname from the default configuration file or the DHCP reply, the device reads the configuration file that has the same name as its hostname (*hostname-config* or *hostname.cfg*, depending on whether *network-config* or *cisconet.cfg* was read earlier) from the TFTP server. If the *cisconet.cfg* file is read, the filename of the host is truncated to eight characters.

If the device cannot read the *network-config*, *cisconet.cfg*, or the hostname file, it reads the *router-config* file. If the device cannot read the *router-config* file, it reads the *ciscotr.cfg* file.



Note The device broadcasts TFTP server requests if the TFTP server is not obtained from the DHCP replies, if all attempts to read the configuration file through unicast transmissions fail, or if the TFTP server name cannot be resolved to an IP address.

How to Control Environment Variables

With a normally operating device, you enter the boot loader mode only through the console connection. Unplug the switch power cord, then reconnect the power cord. Hold down the **MODE** button until you see the boot loader switch prompt

The device boot loader software provides support for nonvolatile environment variables, which can be used to control how the boot loader or any other software running on the system, functions. Boot loader environment variables are similar to environment variables that can be set on UNIX or DOS systems.

Environment variables that have values are stored in flash memory outside of the flash file system.

Each line in these files contains an environment variable name and an equal sign followed by the value of the variable. A variable has no value if it is not present; it has a value if it is listed even if the value is a null string. A variable that is set to a null string (for example, “”) is a variable with a value. Many environment variables are predefined and have default values.

Environment variables store two kinds of data:

- Data that controls code, which does not read the Cisco IOS configuration file. For example, the name of a boot loader helper file, which extends or patches the functionality of the boot loader can be stored as an environment variable.
- Data that controls code, which is responsible for reading the Cisco IOS configuration file. For example, the name of the Cisco IOS configuration file can be stored as an environment variable.

You can change the settings of the environment variables by accessing the boot loader or by using Cisco IOS commands. Under normal circumstances, it is not necessary to alter the setting of the environment variables.

Common Environment Variables

This table describes the function of the most common environment variables.

Table 158: Common Environment Variables

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
BOOT	<p>set BOOT <i>filesystem</i> :<i>/ file-url ...</i></p> <p>A semicolon-separated list of executable files to try to load and execute when automatically booting. If the BOOT environment variable is not set, the system attempts to load and execute the first executable image it can find by using a recursive, depth-first search through the flash file system. If the BOOT variable is set but the specified images cannot be loaded, the system attempts to boot the first bootable file that it can find in the flash file system.</p>	<p>boot system <i>{filesystem : /file-url ...</i></p> <p>Specifies the Cisco IOS image to load during the next boot cycle and the stack members on which the image is loaded. This command changes the setting of the BOOT environment variable.</p>
MANUAL_BOOT	<p>set MANUAL_BOOT yes</p> <p>Decides whether the switch automatically or manually boots.</p> <p>Valid values are 1, yes, 0, and no. If it is set to no or 0, the boot loader attempts to automatically boot up the system. If it is set to anything else, you must manually boot up the switch from the boot loader mode.</p>	<p>boot manual</p> <p>Enables manually booting the switch during the next boot cycle and changes the setting of the MANUAL_BOOT environment variable.</p> <p>The next time you reboot the system, the switch is in boot loader mode. To boot up the system, use the boot flash: <i>filesystem : / file-url</i> boot loader command, and specify the name of the bootable image.</p>

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
CONFIG_FILE	<p>set CONFIG_FILE flash:/ file-url</p> <p>Changes the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration.</p>	<p>boot config-file flash:/ file-url</p> <p>Specifies the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration. This command changes the CONFIG_FILE environment variable.</p>
SWITCH_NUMBER	<p>set SWITCH_NUMBER stack-member-number</p> <p>Changes the member number of a stack member.</p>	<p>switch current-stack-member-number renumber new-stack-member-number</p> <p>Changes the member number of a stack member.</p>
SWITCH_PRIORITY	<p>set SWITCH_PRIORITY stack-member-number</p> <p>Changes the priority value of a stack member.</p>	<p>switch stack-member-number priority priority-number</p> <p>Changes the priority value of a stack member.</p>
BAUD	<p>set BAUD baud-rate</p>	<p>line console 0 speed speed-value</p> <p>Configures the baud rate.</p>
ENABLE_BREAK	<p>set ENABLE_BREAK yes/no</p>	<p>boot enable-break switch yes/no</p> <p>This command can be issued when the flash filesystem is initialized when ENABLE_BREAK is set to yes.</p>

Environment Variables for TFTP

When the switch is connected to a PC through the Ethernet management port, you can download or upload a configuration file to the boot loader by using TFTP. Make sure the environment variables in this table are configured.

Table 159: Environment Variables for TFTP

Variable	Description
MAC_ADDR	Specifies the MAC address of the switch. Note We recommend that you do not modify this variable. However, if you modify this variable after the boot loader is up or the value is different from the saved value, enter this command before using TFTP. A reset is required for the new value to take effect.
IP_ADDRESS	Specifies the IP address and the subnet mask for the associated IP subnet of the switch.
DEFAULT_ROUTER	Specifies the IP address and subnet mask of the default gateway.

Scheduled Reload of the Software Image

You can schedule a reload of the software image to occur on the device at a later time (for example, late at night or during the weekend when the device is used less), or you can synchronize a reload network-wide (for example, to perform a software upgrade on all devices in the network).



Note A scheduled reload must take place within approximately 24 days.

You have these reload options:

- Reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 hours. You can specify the reason for the reload in a string up to 255 characters in length.
- Reload of the software to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight.

The **reload** command halts the system. If the system is not set to manually boot up, it reboots itself.

If your device is configured for manual booting, do not reload it from a virtual terminal. This restriction prevents the device from entering the boot loader mode and then taking it from the remote user's control.

If you modify your configuration file, the device prompts you to save the configuration before reloading. During the save operation, the system requests whether you want to proceed with the save if the CONFIG_FILE environment variable points to a startup configuration file that no longer exists. If you proceed in this situation, the system enters setup mode upon reload.

To cancel a previously scheduled reload, use the **reload cancel** privileged EXEC command.

How to Perform Device Setup Configuration

Using DHCP to download a new image and a new configuration to a device requires that you configure at least two devices. One device acts as a DHCP and TFTP server and the second device (client) is configured to download either a new configuration file or a new configuration file and a new image file.

Configuring DHCP Autoconfiguration (Only Configuration File)

This task describes how to configure DHCP autoconfiguration of the TFTP and DHCP settings on an existing device in the network so that it can support the autoconfiguration of a new device.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **exit**
8. **tftp-server flash:***filename.text*
9. **interface** *interface-id*
10. **no switchport**
11. **ip address** *address mask*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	ip dhcp pool <i>poolname</i> Example: <pre>Switch(config)# ip dhcp pool pool</pre>	Creates a name for the DHCP server address pool, and enters DHCP pool configuration mode.
Step 3	boot <i>filename</i> Example:	Specifies the name of the configuration file that is used as a boot image.

	Command or Action	Purpose
	Switch(dhcp-config)# boot config-boot.text	
Step 4	network <i>network-number mask prefix-length</i> Example: Switch(dhcp-config)# network 10.10.10.0 255.255.255.0	Specifies the subnet network number and mask of the DHCP address pool. Note The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).
Step 5	default-router <i>address</i> Example: Switch(dhcp-config)# default-router 10.10.10.1	Specifies the IP address of the default router for a DHCP client.
Step 6	option 150 <i>address</i> Example: Switch(dhcp-config)# option 150 10.10.10.1	Specifies the IP address of the TFTP server.
Step 7	exit Example: Switch(dhcp-config)# exit	Returns to global configuration mode.
Step 8	tftp-server flash: <i>filename.text</i> Example: Switch(config)# tftp-server flash:config-boot.text	Specifies the configuration file on the TFTP server.
Step 9	interface <i>interface-id</i> Example:	Specifies the address of the client that will receive the configuration file.
Step 10	no switchport Example: Switch(config-if)# no switchport	Puts the interface into Layer 3 mode.
Step 11	ip address <i>address mask</i> Example:	Specifies the IP address and mask for the interface.

	Command or Action	Purpose
	Switch(config-if)# ip address 10.10.10.1 255.255.255.0	
Step 12	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.

Configuring DHCP Auto-Image Update (Configuration File and Image)

This task describes DHCP autoconfiguration to configure TFTP and DHCP settings on an existing device to support the installation of a new switch.

Before you begin

You must first create a text file (for example, `autoinstall_dhcp`) that will be uploaded to the device. In the text file, put the name of the image that you want to download (for example, `c3750e-ipservices-mz.122-44.3.SE.tar` or `c3750x-ipservices-mz.122-53.3.SE2.tar`). This image must be a tar and not a bin file.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **option 125** *hex*
8. **copy tftp flash** *filename.txt*
9. **copy tftp flash** *imagename.bin*
10. **exit**
11. **tftp-server flash:** *config.text*
12. **tftp-server flash:** *imagename.bin*
13. **tftp-server flash:** *filename.txt*
14. **interface** *interface-id*
15. **no switchport**
16. **ip address** *address mask*
17. **end**
18. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ip dhcp pool <i>poolname</i> Example: Switch(config)# ip dhcp pool pool1	Creates a name for the DHCP server address pool and enter DHCP pool configuration mode.
Step 3	boot <i>filename</i> Example: Switch(dhcp-config)# boot config-boot.text	Specifies the name of the file that is used as a boot image.
Step 4	network <i>network-number mask prefix-length</i> Example: Switch(dhcp-config)# network 10.10.10.0 255.255.255.0	Specifies the subnet network number and mask of the DHCP address pool. Note The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).
Step 5	default-router <i>address</i> Example: Switch(dhcp-config)# default-router 10.10.10.1	Specifies the IP address of the default router for a DHCP client.
Step 6	option 150 <i>address</i> Example: Switch(dhcp-config)# option 150 10.10.10.1	Specifies the IP address of the TFTP server.
Step 7	option 125 <i>hex</i> Example: Switch(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370	Specifies the path to the text file that describes the path to the image file.

	Command or Action	Purpose
Step 8	copy tftp flash <i>filename.txt</i> Example: Switch(config)# copy tftp flash image.bin	Uploads the text file to the device.
Step 9	copy tftp flash <i>imagename.bin</i> Example: Switch(config)# copy tftp flash image.bin	Uploads the tar file for the new image to the device.
Step 10	exit Example: Switch(dhcp-config)# exit	Returns to global configuration mode.
Step 11	tftp-server flash: <i>config.txt</i> Example: Switch(config)# tftp-server flash:config-boot.txt	Specifies the Cisco IOS configuration file on the TFTP server.
Step 12	tftp-server flash: <i>imagename.bin</i> Example: Switch(config)# tftp-server flash:image.bin	Specifies the image name on the TFTP server.
Step 13	tftp-server flash: <i>filename.txt</i> Example: Switch(config)# tftp-server flash:boot-config.txt	Specifies the text file that contains the name of the image file to download
Step 14	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet 1/0/4	Specifies the address of the client that will receive the configuration file.
Step 15	no switchport Example: Switch(config-if)# no switchport	Puts the interface into Layer 3 mode.

	Command or Action	Purpose
Step 16	ip address <i>address mask</i> Example: <pre>Switch(config-if)# ip address 10.10.10.1 255.255.255.0</pre>	Specifies the IP address and mask for the interface.
Step 17	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 18	copy running-config startup-config Example: <pre>Switch(config-if)# end</pre>	(Optional) Saves your entries in the configuration file.

Configuring the Client to Download Files from DHCP Server



Note You should only configure and enable the Layer 3 interface. Do not assign an IP address or DHCP-based autoconfiguration with a saved configuration.

SUMMARY STEPS

1. **configure terminal**
2. **boot host dhcp**
3. **boot host retry timeout** *timeout-value*
4. **banner config-save** ^C *warning-message* ^C
5. **end**
6. **show boot**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	boot host dhcp Example:	Enables autoconfiguration with a saved configuration.

	Command or Action	Purpose
	<code>Switch(conf)# boot host dhcp</code>	
Step 3	boot host retry timeout <i>timeout-value</i> Example: <code>Switch(conf)# boot host retry timeout 300</code>	(Optional) Sets the amount of time the system tries to download a configuration file. Note If you do not set a timeout, the system will try indefinitely to obtain an IP address from the DHCP server.
Step 4	banner config-save ^C <i>warning-message</i> ^C Example: <code>Switch(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot^C</code>	(Optional) Creates warning messages to be displayed when you try to save the configuration file to NVRAM.
Step 5	end Example: <code>Switch(config-if)# end</code>	Returns to privileged EXEC mode.
Step 6	show boot Example: <code>Switch# show boot</code>	Verifies the configuration.

Manually Assigning IP Information to Multiple SVIs

This task describes how to manually assign IP information to multiple switched virtual interfaces (SVIs):

SUMMARY STEPS

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **exit**
5. **ip default-gateway** *ip-address*
6. **end**
7. **show interfaces vlan** *vlan-id*
8. **show ip redirects**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface vlan <i>vlan-id</i> Example: Switch(config)# interface vlan 99	Enters interface configuration mode, and enter the VLAN to which the IP information is assigned. The range is 1 to 4094.
Step 3	ip address <i>ip-address subnet-mask</i> Example: Switch(config-vlan)# ip address 10.10.10.2 255.255.255.0	Enters the IP address and subnet mask.
Step 4	exit Example: Switch(config-vlan)# exit	Returns to global configuration mode.
Step 5	ip default-gateway <i>ip-address</i> Example: Switch(config)# ip default-gateway 10.10.10.1	<p>Enters the IP address of the next-hop router interface that is directly connected to the device where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the device.</p> <p>Once the default gateway is configured, the device has connectivity to the remote networks with which a host needs to communicate.</p> <p>Note When your device is configured to route with IP, it does not need to have a default gateway set.</p>
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show interfaces vlan <i>vlan-id</i> Example:	Verifies the configured IP address.

	Command or Action	Purpose
	Switch# <code>show interfaces vlan 99</code>	
Step 8	show ip redirects Example: Switch# <code>show ip redirects</code>	Verifies the configured default gateway.

Configuring the NVRAM Buffer Size

The default NVRAM buffer size is 512 KB. In some cases, the configuration file might be too large to save to NVRAM. Typically, this occurs when you have many switches in a switch stack. You can configure the size of the NVRAM buffer to support larger configuration files. The new NVRAM buffer size is synced to all current and new member switches.



Note After you configure the NVRAM buffer size, reload the switch or switch stack.

When you add a switch to a stack and the NVRAM size differs, the new switch syncs with the stack and reloads automatically.

SUMMARY STEPS

1. `configure terminal`
2. `boot buffersize size`
3. `end`
4. `show boot`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	boot buffersize size Example: Switch(config)# <code>boot buffersize 524288</code>	Configures the NVRAM buffersize in KB. The valid range for <i>size</i> is from 4096 to 1048576.
Step 3	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 4	show boot Example: Switch# show boot	Verifies the configuration.

Modifying the Device Startup Configuration

Specifying the Filename to Read and Write the System Configuration

By default, the Cisco IOS software uses the `config.text` file to read and write a nonvolatile copy of the system configuration. However, you can specify a different filename, which will be loaded during the next boot cycle.

Before you begin

Use a standalone device for this task.

SUMMARY STEPS

1. **configure terminal**
2. **boot config-file** file name
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	boot config-file file name Example: Switch(config)# boot config-file config.text	Specifies the configuration file to load during the next boot cycle. <i>file-url</i> —The path (directory) and the configuration filename. Filenames and directory names are case-sensitive.
Step 3	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 4	show boot Example: Switch# show boot	Verifies your entries. The boot global configuration command changes the setting of the CONFIG_FILE environment variable.
Step 5	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Manually Booting the Switch

By default, the switch automatically boots up; however, you can configure it to manually boot up.

Before you begin

Use a standalone switch for this task.

SUMMARY STEPS

1. **configure terminal**
2. **boot manual**
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	boot manual Example: Switch(config)# boot manual	Enables the switch to manually boot up during the next boot cycle.
Step 3	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 4	show boot Example: Switch# show boot	Verifies your entries. The boot manual global command changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode, shown by the <i>switch:</i> prompt. To boot up the system, use the boot filesystem:/file-url boot loader command. <ul style="list-style-type: none"> <i>filesystem:</i>—Uses flash: for the system board flash device. Switch: boot flash: For <i>file-url</i>—Specifies the path (directory) and the name of the bootable image. Filenames and directory names are case-sensitive.
Step 5	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Scheduled Software Image Reload

This task describes how to configure your device to reload the software image at a later time.

SUMMARY STEPS

1. **configure terminal**
2. **copy running-config startup-config**
3. **reload in** *[hh:]mm* *[text]*
4. **reload at** *hh: mm* *[month day | day month]* *[text]*
5. **reload cancel**
6. **show reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	copy running-config startup-config Example: <pre>copy running-config startup-config</pre>	Saves your device configuration information to the startup configuration before you use the reload command.
Step 3	reload in [hh:]mm [text] Example: <pre>Switch(config)# reload in 12 System configuration has been modified. Save? [yes/no]: y</pre>	Schedules a reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days. You can specify the reason for the reload in a string up to 255 characters in length.
Step 4	reload at hh: mm [month day day month] [text] Example: <pre>Switch(config)# reload at 14:00</pre>	Specifies the time in hours and minutes for the reload to occur. Note Use the at keyword only if the device system clock has been set (through Network Time Protocol (NTP), the hardware calendar, or manually). The time is relative to the configured time zone on the device. To schedule reloads across several devices to occur simultaneously, the time on each device must be synchronized with NTP.
Step 5	reload cancel Example: <pre>Switch(config)# reload cancel</pre>	Cancels a previously scheduled reload.
Step 6	show reload Example: <pre>show reload</pre>	Displays information about a previously scheduled reload or identifies if a reload has been scheduled on the device.

Monitoring Device Setup Configuration

Example: Verifying the Device Running Configuration

```
Switch# show running-config
Building configuration...

Current configuration: 1363 bytes
!
version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
```

```

no service password-encryption
!
hostname Stack1
!
enable secret 5 $1$ej9.$DMUvAUUnZOAmvmgqBEzIxEO
!
.
<output truncated>
.
interface gigabitethernet6/0/2
mvr type source

<output truncated>

...!
interface VLAN1
 ip address 172.20.137.50 255.255.255.0
 no ip directed-broadcast
!
 ip default-gateway 172.20.137.1 !
!
 snmp-server community private RW
 snmp-server community public RO
 snmp-server community private@es0 RW
 snmp-server community public@es0 RO
 snmp-server chassis-id 0x12
!
end

```

Examples: Displaying Software Install

This example displays software bootup in install mode:

```
switch# boot flash:/c3560cx-universalk9-mz.152-3.E/c3560cx-universalk9-tar.152-3.E.bin
```

Configuration Examples for Performing Device Setup

Example: Configuring a Device as a DHCP Server

```

Switch# configure terminal
Switch(config)# ip dhcp pool pool1
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# boot config-boot.text
Switch(dhcp-config)# default-router 10.10.10.1
Switch(dhcp-config)# option 150 10.10.10.1
Switch(dhcp-config)# exit
Switch(config)# tftp-server flash:config-boot.text
Switch(config)# interface gigabitethernet 1/0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1 255.255.255.0
Switch(config-if)# end

```

Example: Configuring DHCP Auto-Image Update

Example: Configuring a Device to Download Configurations from a DHCP Server

This example uses a Layer 3 SVI interface on VLAN 99 to enable DHCP-based autoconfiguration with a saved configuration:

```
Switch# configure terminal
Switch(config)# boot host dhcp
Switch(config)# boot host retry timeout 300
Switch(config)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause
  You to No longer Automatically Download Configuration Files at Reboot^C
Switch(config)# vlan 99
Switch(config-vlan)# interface vlan 99
Switch(config-if)# no shutdown
Switch(config-if)# end
Switch# show boot
BOOT path-list:
Config file:          flash:/config.text
Private Config file:  flash:/private-config.text
Enable Break:        no
Manual Boot:         no
HELPER path-list:
NVRAM/Config file
  buffer size:       32768
Timeout for Config
  Download:          300 seconds
Config Download
  via DHCP:         enabled (next boot: enabled)
Switch#
```

Example: Configuring NVRAM Buffer Size

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# boot buffersize 600000
Switch(config)# end
Switch# show boot
BOOT path-list      :
Config file         : flash:/config.text
Private Config file : flash:/private-config.text
Enable Break       : no
Manual Boot        : no
HELPER path-list   :
Auto upgrade       : yes
Auto upgrade path  :
NVRAM/Config file
  buffer size:     600000
Timeout for Config
  Download:       300 seconds
Config Download
  via DHCP:      enabled (next boot: enabled)
Switch#
```




CHAPTER 78

Configuring RTU Licenses

- [Finding Feature Information, on page 1617](#)
- [Restrictions for Configuring RTU Licenses, on page 1617](#)
- [Information About Configuring RTU Licenses, on page 1618](#)
- [How to Configure RTU Licenses, on page 1619](#)
- [Monitoring and Maintaining RTU Licenses, on page 1622](#)
- [Configuration Examples for RTU Licensing, on page 1623](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restrictions for Configuring RTU Licenses

The following are the restrictions for configuring and using RTU licenses.

- AP count licenses can be ordered and pre-activated on your switch.
- Imaged based licenses can be upgraded. AP count licenses can be deactivated and moved between switches and controllers.
- To activate a license, you must reboot your switch after configuring the new license level. The AP-count license does not require a reboot to activate.
- An expired evaluation license can not be reactivated after reboot.
- Stack members of a switch stack must run the same license level. If the license level is different, the switch will not join the stack until it is changed and rebooted from the active switch of the stack.
- Adder AP-count licenses are installed in the factory.

Information About Configuring RTU Licenses

Right-To-Use Licensing

Right-to-use (RTU) licensing allows you to order and activate a specific license type and level, and then to manage license usage on your switch. The types of licenses available to order by duration are:

- Permanent licenses—Purchased with a specific feature set with no expiration date.
- Evaluation licenses—Pre-installed on the switch and is valid for only a 90 day in-use period.

To activate a permanent or evaluation license, you are required to accept the End-User License Agreement (EULA).

A permanent license can be moved from one device to another. To activate a license, you must reboot your switch.

If you activate the evaluation license, it will expire in 90 days. An evaluation license is a manufacturing image on your switch and is not transferable to another switch. Once activated, this type of license cannot be deactivated until it expires. After your evaluation period expires, at the next reload your switch image will return to its default license and network operations are not impacted.

Right-To-Use Image-Based Licenses

Right-to-use image licenses support a set of features based on a specific image-based license:

- LAN Base—Layer 2 features.
- IP Base—Layer 2 and Layer 3 features.
- IP Services—Layer 2, Layer 3, and IPv6 features. (Applicable only to switches and not controllers.)

The default image license for the switches is as follows:

- Catalyst 2960-CX switches: LAN Base
- Catalyst 3560-CX switches: IP Base

Right-To-Use License States

After you configure a specific license type and level, you can manage your licenses by monitoring the license state.

Table 160: RTU License States

License State	Description
Active, In Use	EULA was accepted and the license is in use after device reboot.
Active, Not In Use	EULA was accepted and the switch is ready to use when the license is enabled.

License State	Description
Not Activated	EULA was not accepted.

Guidelines to follow when monitoring your image based license state:

- A purchased permanent license is set to *Active, In Use* state only after a switch reboot.
- If more than one license was purchased, a reboot will activate the license with the highest feature set. For instance, the IP Services license is activated and not the LAN Base license.
- Remaining licenses purchased after switch reboot, stay in **Active, Not In Use** state.



Note For the AP count license, to change the state to Active, In Use, you must first make sure that the evaluation AP count license is deactivated.

Mobility Controller Mode

AP-count licenses are used only when the switch is in Mobility Controller mode. The MC is the gatekeeper for tracking the AP-count licenses and allows an access point to join or not.

Management of AP-count licenses is performed by the in mobility controller mode configurable through the CLI.

Right-To-Use Adder AP-Count Rehosting Licenses

Revoking a license from one device and installing it on another is called rehosting. You might want to rehost a license to change the purpose of a device.

To rehost a license, you must deactivate the adder ap-count license from one device and activate the same license on another device.

Evaluation licenses cannot be rehosted.

How to Configure RTU Licenses

Activating an Image Based License

To activate image based licenses, complete the following task:

SUMMARY STEPS

1. `license right-to-use activate { ipbase | ipservices | lanbase } [all | evaluation | slot slot-number] [acceptEULA]`
2. `reload [LINE | at | cancel | in | slot stack-member-number | standby-cpu]`
3. `show license right-to-use usage [slot slot-number]`

DETAILED STEPS

	Command or Action	Purpose																																				
Step 1	<p>license right-to-use activate { ipbase ipservices lanbase } [all evaluation slot slot-number] [acceptEULA]</p> <p>Example:</p> <pre>Switch# license right-to-use activate ipservices all acceptEULA</pre>	<p>Activates the license level. Activation can happen on all switches and also include the EULA acceptance.</p> <p>Note If you do not accept EULA, the modified configuration will not take effect after reload. The default license (or a license that was not deactivated) becomes active after reload.</p>																																				
Step 2	<p>reload [LINE at cancel in slot stack-member-number standby-cpu]</p> <p>Example:</p> <pre>Switch# reload slot 1 Proceed with reload? [confirm] y</pre>	<p>Reloads a specific stack member to complete the activation process for the RTU adder AP-count license.</p> <p>Note The reminder to accept the EULA is displayed after reload if it was not accepted earlier.</p> <p>When changing license level, you are not required to save the configuration. But, it is a good practice to ensure all the configuration is stored properly before reload. Changing from a higher license level to a lower license level on reboot will remove CLIs that are not applicable. Ensure that all features in the lower license level that are actively used are not removed.</p>																																				
Step 3	<p>show license right-to-use usage [slot slot-number]</p> <p>Example:</p> <pre>Switch# show license right-to-use usage</pre> <table border="1"> <thead> <tr> <th>Slot#</th> <th>License Name</th> <th>Type</th> </tr> <tr> <th>usage-duration(y:m:d)</th> <th>In-Use</th> <th>EULA</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>ipservices</td> <td>Permanent</td> </tr> <tr> <td>0 :10:27</td> <td>yes yes</td> <td></td> </tr> <tr> <td>1</td> <td>ipservices</td> <td>Evaluation</td> </tr> <tr> <td>0 :0 :0</td> <td>no no</td> <td></td> </tr> <tr> <td>1</td> <td>ipbase</td> <td>Permanent</td> </tr> <tr> <td>0 :0 :9</td> <td>no yes</td> <td></td> </tr> <tr> <td>1</td> <td>ipbase</td> <td>Evaluation</td> </tr> <tr> <td>0 :0 :0</td> <td>no no</td> <td></td> </tr> <tr> <td>1</td> <td>lanbase</td> <td>Permanent</td> </tr> <tr> <td>0 :11:12</td> <td>no yes</td> <td></td> </tr> </tbody> </table> <pre>Switch#</pre>	Slot#	License Name	Type	usage-duration(y:m:d)	In-Use	EULA	1	ipservices	Permanent	0 :10:27	yes yes		1	ipservices	Evaluation	0 :0 :0	no no		1	ipbase	Permanent	0 :0 :9	no yes		1	ipbase	Evaluation	0 :0 :0	no no		1	lanbase	Permanent	0 :11:12	no yes		<p>Displays detailed usage information.</p>
Slot#	License Name	Type																																				
usage-duration(y:m:d)	In-Use	EULA																																				
1	ipservices	Permanent																																				
0 :10:27	yes yes																																					
1	ipservices	Evaluation																																				
0 :0 :0	no no																																					
1	ipbase	Permanent																																				
0 :0 :9	no yes																																					
1	ipbase	Evaluation																																				
0 :0 :0	no no																																					
1	lanbase	Permanent																																				
0 :11:12	no yes																																					

Activating an AP-Count License

SUMMARY STEPS

1. `license right-to-use activate {apcount ap-number slot slot-num} | evaluation} [acceptEULA]`
2. `show license right-to-use usage [slot slot-number]`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>license right-to-use activate {apcount ap-number slot slot-num} evaluation} [acceptEULA]</code></p> <p>Example:</p> <pre>Switch# license right to use activate apcount 5 slot 1 acceptEULA</pre>	Activates one or more adder AP-count licenses and immediately accepts the EULA.
Step 2	<p><code>show license right-to-use usage [slot slot-number]</code></p> <p>Example:</p> <pre>Switch# show license right-to-use usage Slot# License Name Type usage-duration(y:m:d) In-Use EULA 1 ipservices permanent 0 :3 :29 yes yes 1 ipservices evaluation 0 :0 :0 no no 1 ipbase permanent 0 :0 :0 no no 1 ipbase evaluation 0 :0 :0 no no 1 lanbase permanent 0 :0 :0 no no 1 apcount evaluation 0 :3 :11 no no 1 apcount base 0 :0 :0 no yes 1 apcount adder 0 :0 :17 yes yes Switch#</pre>	Displays detailed usage information.

Obtaining an Upgrade or Capacity Adder License

You can use the capacity adder licenses to increase the number of access points supported by the device.

SUMMARY STEPS

1. `license right-to-use {activate | deactivate} apcount {ap-number | evaluation} slot slot-num [acceptEULA]`

DETAILED STEPS

	Command or Action	Purpose
Step 1	license right-to-use {activate deactivate} apcount {ap-number evaluation} slot slot-num [acceptEULA] Example: <pre>Switch# license right to use activate apcount 5 slot 2 acceptEULA</pre>	Activates one or more adder AP-count licenses and immediately accepts the EULA.

Rehosting a License

To rehost a license, you have to deactivate the license from one device and then activate the same license on another device.

SUMMARY STEPS

1. `license right-to-use deactivate [license-level] apcount ap-number slot slot-num`
2. `license right-to-use activate [license-level] slot slot-num [acceptEULA]`

DETAILED STEPS

	Command or Action	Purpose
Step 1	license right-to-use deactivate [license-level] apcount ap-number slot slot-num Example: <pre>Switch# license right-to-use deactivate apcount 1 slot 1</pre>	Deactivates the license on one device. The IP Base license level is deactivate from slot 1 in the example here.
Step 2	license right-to-use activate [license-level] slot slot-num [acceptEULA] Example: <pre>Switch# license right to use activate ipbase slot 2 acceptEULA</pre>	Activates the license on another device. The IP Base license level is rehosted on slot 2 in the example here.

Monitoring and Maintaining RTU Licenses

Command	Purpose
<code>show license right-to-use default</code>	Displays the default license information.
<code>show license right-to-use detail</code>	Displays detailed information of all the licenses in the switch stack.

Command	Purpose
<code>show license right-to-use eula {evaluation permanent}</code>	Displays the end user license agreement.
<code>show license right-to-use mismatch</code>	Displays the license information that does not match.
<code>show license right-to-use slot <i>slot-number</i></code>	Displays the license information for a specific slot in a switch stack.
<code>show license right-to-use summary</code>	Displays a summary of the license information on the entire switch stack.
<code>show license right-to-use usage [slot <i>slot-number</i>]</code>	Displays detailed information about usage for all licenses in the switch stack.
<code>show switch</code>	Displays detailed information of every member in a switch stack including the state of the license.

Configuration Examples for RTU Licensing

Examples: Activating RTU Image Based Licenses

This example shows how to activate an IP Services image license and accept the EULA for a specific slot:

```
Switch# license right-to-use activate ipservices slot 1 acceptEULA
% switch-1:stack-mgr:Reboot the switch to invoke the highest activated License level
```

This example shows how to activate a license for evaluation:

```
Switch# license right-to-use activate ipservices evaluation acceptEULA
% switch-1:stack-mgr:Reboot the switch to invoke the highest activated License level
```

Examples: Displaying RTU Licensing Information

Example: Displaying RTU License Details

This example shows all the detailed information for the RTU licenses on slot 1:

Example: Displaying RTU License Mismatch

This example shows the license information of the switches in a stack and a mismatch state of a member switch. The member must match the active.

```
Switch# show switch
```

```
Switch/Stack Mac Address : 1c1d.8625.7700 - Local Mac Address
                                     H/W   Current
Switch#   Role       Mac Address      Priority Version  State
-----
*1        Active    1c1d.8625.7700   15      V02     Ready
2         Standby   bc16.f55c.ab80   7       V04     Ready
3         Member    580a.2095.da00   1       V03     Lic-Mismatch
```



Note To resolve the license mismatch, first check the RTU license summary:

```
Switch# show license right-to-use
```

Then change the license level of the mismatched switched so that it is the same license level of the active switch. This example shows that the IP Base license was activated for the member switch to match the active switch.

```
Switch# license right-to-use activate ipbase slot 3 acceptEULA
```

Example: Displaying RTU Licensing Usage



CHAPTER 79

Clustering Switches

- [Understanding Switch Clusters, on page 1625](#)
- [Planning a Switch Cluster, on page 1627](#)
- [Using the CLI to Manage Switch Clusters, on page 1636](#)
- [Using SNMP to Manage Switch Clusters, on page 1637](#)

Understanding Switch Clusters

A *switch cluster* is a set of up to 16 connected, cluster-capable Catalyst switches that are managed as a single entity. The Switch in the cluster use the Switch clustering technology so that you can configure and troubleshoot a group of different Catalyst desktop Switch platforms through a single IP address.

In a Switch cluster, 1 Switch must be the *cluster command* Switch and up to 15 other Switch can be *cluster member switches*. The total number of Switch in a cluster cannot exceed 16 Switch. The cluster command Switch is the single point of access used to configure, manage, and monitor the cluster member Switch. Cluster members can belong to only one cluster at a time.



Note A Switch cluster is different from a *switch stack*. A switch stack is a set of Catalyst 3750-X, Catalyst 3750-E, or Catalyst 3750 Switch connected through their stack ports.

The benefits of clustering Switch include:

- Management of Catalyst Switch regardless of their interconnection media and their physical locations. The Switch can be in the same location, or they can be distributed across a Layer 2 or Layer 3 (if your cluster is using a Catalyst 3560, Catalyst 3750, Catalyst 3560-E, Catalyst 3750-E, Catalyst 3560-X, or Catalyst 3750-X Switch as a Layer 3 router between the Layer 2 Switch in the cluster) network.
- Command-switch redundancy if a cluster command Switch fails. One or more Switch can be designated as *standby cluster command switches* to avoid loss of contact with cluster members. A *cluster standby group* is a group of standby cluster command Switch.
- Management of a variety of Catalyst Switch through a single IP address. This conserves on IP addresses, especially if you have a limited number of them. All communication with the switch cluster is through the cluster command Switch IP address.

The below table lists the Catalyst switches eligible for Switch clustering, including which ones can be cluster command switches and which ones can only be cluster member switches, and the required software versions.

Table 161: Switch Software and Cluster Capability

Switch	Cisco IOS Release	Cluster Capability
Catalyst 3750-X	12.2(53)SE2 or later	Member or command switch
Catalyst 3750-E	12.2(35)SE2 or later	Member or command switch
Catalyst 3750	12.1(11)AX or later	Member or command switch
Catalyst 3560-X	12.2(53)SE1 or later	Member or command switch
Catalyst 3560-E	12.2(35)SE2 or later	Member or command switch
Catalyst 3560	12.1(19)EA1b or later	Member or command switch
Catalyst 3550	12.1(4)EA1 or later	Member or command switch
Catalyst 2970	12.1(11)AX or later	Member or command switch
Catalyst 2960	12.2(25)FX or later	Member or command switch
Catalyst 2955	12.1(12c)EA1 or later	Member or command switch
Catalyst 2950	12.0(5.2)WC(1) or later	Member or command switch
Catalyst 2950 LRE	12.1(11)JY or later	Member or command switch
Catalyst 2940	12.1(13)AY or later	Member or command switch
Catalyst 3500 XL	12.0(5.1)XU or later	Member or command switch
Catalyst 2900 XL (8-MB switches)	12.0(5.1)XU or later	Member or command switch
Catalyst 2900 XL (4-MB switches)	11.2(8.5)SA6 (recommended)	Member switch only
Catalyst 1900 and 2820	9.00(-A or -EN) or later	Member switch only

Cluster Command Switch Characteristics

A cluster command Switch must meet these requirements:

- It is running a supported software release.
- It has an IP address.
- It has Cisco Discovery Protocol (CDP) Version 2 enabled (the default).
- It is not a command or cluster member Switch of another cluster.
- It is connected to the standby cluster command Switch through the management VLAN and to the cluster member Switch through a common VLAN.

Standby Cluster Command Switch Characteristics

A standby cluster command Switch must meet these requirements:

- It is running a supported software release.
- It has an IP address.

- It has CDP Version 2 enabled.
- It is connected to the command Switch and to other standby command Switch through its management VLAN.
- It is connected to all other cluster member Switch (except the cluster command and standby command Switch) through a common VLAN.
- It is redundantly connected to the cluster so that connectivity to cluster member Switch is maintained.
- It is not a command or member Switch of another cluster.



Note Standby cluster command Switch must be the same type of Switch as the cluster command Switch. For example, if the cluster command Switch is a Catalyst 3750-E Switch, the standby cluster command Switch must also be Catalyst 3750-E Switch. See the switch configuration guide of other cluster-capable Switch for their requirements on standby cluster command Switch.

Candidate Switch and Cluster Member Switch Characteristics

Candidate switches are cluster-capable Switch and Switch stacks that have not yet been added to a cluster. Cluster member Switch are switches and switch stacks that have actually been added to a Switch cluster. Although not required, a candidate or cluster member Switch can have its own IP address and password.

To join a cluster, a candidate Switch must meet these requirements:

- It is running cluster-capable software.
- It has CDP Version 2 enabled.
- It is not a command or cluster member Switch of another cluster.
- If a cluster standby group exists, it is connected to every standby cluster command Switch through at least one common VLAN. The VLAN to each standby cluster command Switch can be different.
- The **ip http** server global configuration command must be configured on the Switch.
- It is connected to the cluster command Switch through at least one common VLAN.



Note Catalyst 1900, Catalyst 2820, Catalyst 2900 XL, Catalyst 2940, Catalyst 2950, and Catalyst 3500 XL candidate and cluster member Switch must be connected through their management VLAN to the cluster command switch and standby cluster command switches. For complete information about these Switch in a switch-cluster environment, see the software configuration guide for that specific switch. This requirement does not apply if you have a Catalyst 2960, Catalyst 2970, Catalyst 3550, Catalyst 3560, Catalyst 3560-E, Catalyst 3750, Catalyst 3750-E, Catalyst 3650-X, or Catalyst 3750-X cluster command switch. Candidate and cluster member Switch can connect through any VLAN in common with the cluster command switch.

Planning a Switch Cluster

Anticipating conflicts and compatibility issues is a high priority when you manage several switches through a cluster. This section describes these guidelines, requirements, and caveats that you should understand before you create the cluster.

See the release notes for the list of Catalyst switches eligible for switch clustering, including which ones can be cluster command switches and which ones can only be cluster member switches, and for the required software versions and browser and Java plug-in configurations.

Automatic Discovery of Cluster Candidates and Members

The cluster command switch uses Cisco Discovery Protocol (CDP) to discover cluster member switches, candidate switches, neighboring switch clusters, and edge devices across multiple VLANs and in star or cascaded topologies.



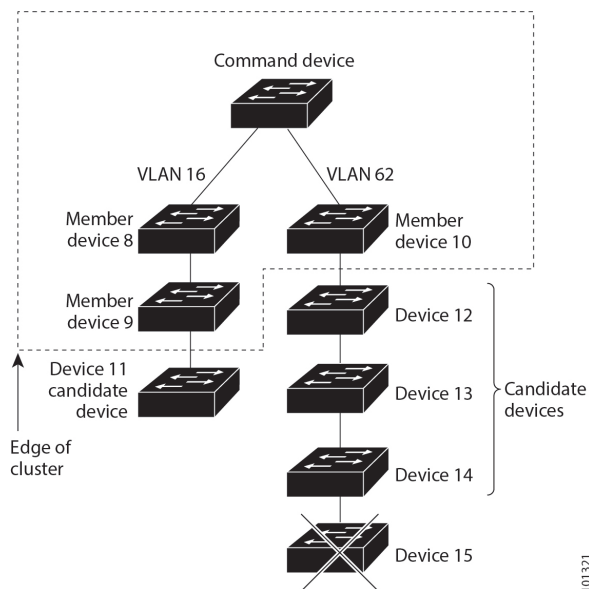
Note Do not disable CDP on the cluster command switch, on cluster members, or on any cluster-capable switches that you might want a cluster command switch to discover.

Discovery Through CDP Hops

By using CDP, a cluster command switch can discover switches up to seven CDP hops away (the default is three hops) from the edge of the cluster. The edge of the cluster is where the last cluster member switches are connected to the cluster and to candidate switches. For example, cluster member switches 9 and 10 in the Figure are at the edge of the cluster.

In the Figure below, the cluster command switch has ports assigned to VLANs 16 and 62. The CDP hop count is three. The cluster command switch discovers switches 11, 12, 13, and 14 because they are within three hops from the edge of the cluster. It does not discover switch 15 because it is four hops from the edge of the cluster.

Figure 123: Discovery Through CDP Hops



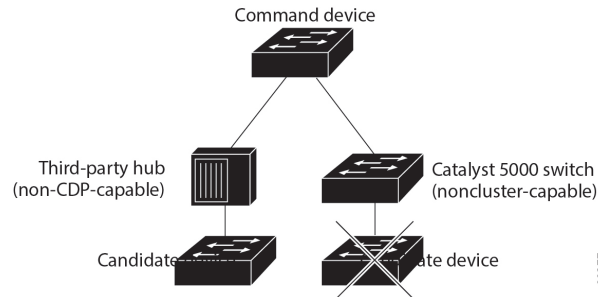
Discovery Through Non-CDP-Capable and Noncluster-Capable Devices

If a cluster command switch is connected to a *non-CDP-capable third-party hub* (such as a non-Cisco hub), it can discover cluster-enabled devices connected to that third-party hub. However, if the cluster command

switch is connected to a *noncluster-capable Cisco device*, it cannot discover a cluster-enabled device connected beyond the noncluster-capable Cisco device.

Below figure shows that the cluster command switch discovers the switch that is connected to a third-party hub. However, the cluster command switch does not discover the switch that is connected to a Catalyst 5000 switch.

Figure 124: Discovery Through Non-CDP-Capable and Noncluster-Capable Devices



Discovery Through Different VLANs

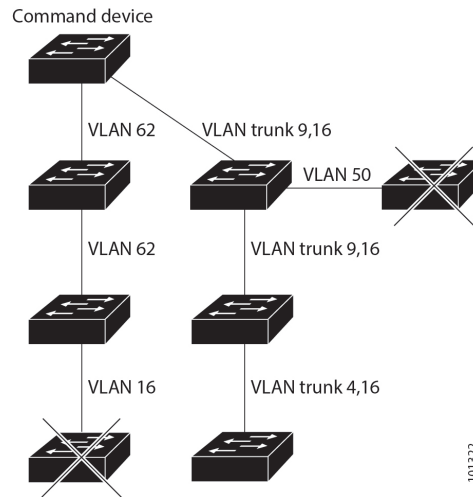
If the cluster command switch is a Catalyst 3560-E, Catalyst 3750-E, Catalyst 3560-X, or Catalyst 3750-X switch, the cluster can have cluster member switches in different VLANs. As cluster member switches, they must be connected through at least one VLAN in common with the cluster command switch. The cluster command switch in the figure as ports assigned to VLANs 9, 16, and 62 and therefore discovers the switches in those VLANs. It does not discover the switch in VLAN 50. It also does not discover the switch in VLAN 16 in the first column because the cluster command switch has no VLAN connectivity to it.

Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL cluster member switches must be connected to the cluster command switch through their management VLAN.



Note For additional considerations about VLANs in switch stacks, see the section “Switch Clusters and Switch Stacks”.

Figure 125: Discovery Through Different VLANs



Discovery Through Different Management VLANs

Catalyst 2960, Catalyst 2970, Catalyst 3550, Catalyst 3560, Catalyst 3560-E, Catalyst 3750, Catalyst 3750-E, Catalyst 3560-X, or Catalyst 3750-X cluster command switches can discover and manage cluster member switches in different VLANs and different management VLANs. As cluster member switches, they must be connected through at least one VLAN in common with the cluster command switch. They do not need to be connected to the cluster command switch through their management VLAN. The default management VLAN is VLAN 1.

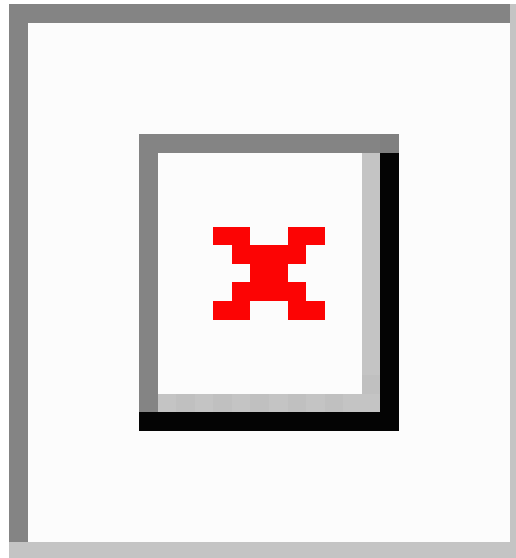


Note If the switch cluster has a Catalyst 3750-E or Catalyst 3750-X switch or switch stack, that switch or switch stack must be the cluster command switch.

The cluster command switch and standby command switch in the figure (assuming they are Catalyst 2960, Catalyst 2970, Catalyst 3550, Catalyst 3560, Catalyst 3560-E, Catalyst 3750, Catalyst 3750-E, Catalyst 3560-X, or Catalyst 3750-X cluster command switches) have ports assigned to VLANs 9, 16, and 62. The management VLAN on the cluster command switch is VLAN 9. Each cluster command switch discovers the switches in the different management VLANs except these:

- Switches 7 and 10 (switches in management VLAN 4) because they are not connected through a common VLAN (meaning VLANs 62 and 9) with the cluster command switch.
- Switch 9 because automatic discovery does not extend beyond a noncandidate device, which is switch 7.

Figure 126: Discovery Through Different Management VLANs with a Layer 3 Cluster Command Switch

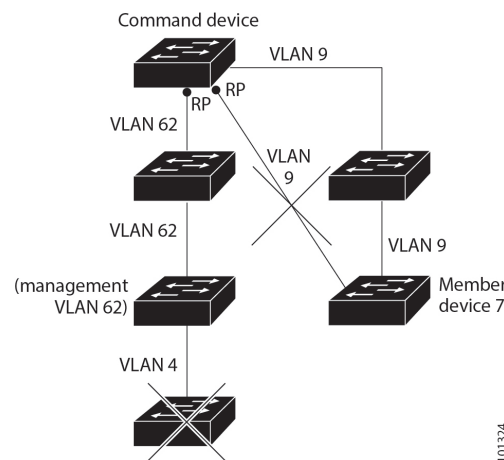


Discovery Through Routed Ports

If the cluster command switch has a routed port (RP) configured, it discovers only candidate and cluster member switches in the *same* VLAN as the routed port.

The Layer 3 cluster command switch in the Figure can discover the switches in VLANs 9 and 62 but not the switch in VLAN 4. If the routed port path between the cluster command switch and cluster member switch 7 is lost, connectivity with cluster member switch 7 is maintained because of the redundant path through VLAN 9.

Figure 127: Discovery Through Routed Ports



Discovery of Newly Installed Switches

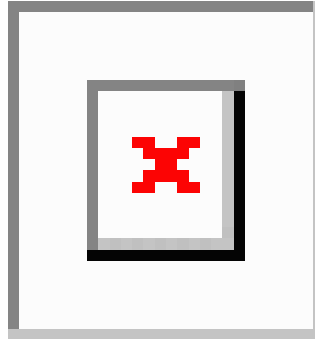
To join a cluster, the new, out-of-the-box switch must be connected to the cluster through one of its access ports. An access port (AP) carries the traffic of and belongs to only one VLAN. By default, the new switch and its access ports are assigned to VLAN 1.

When the new switch joins a cluster, its default VLAN changes to the VLAN of the immediately upstream neighbor. The new switch also configures its access port to belong to the VLAN of the immediately upstream neighbor.

The cluster command switch in the Figure belongs to VLANs 9 and 16. When new cluster-capable switches join the cluster:

- One cluster-capable switch and its access port are assigned to VLAN 9.
- The other cluster-capable switch and its access port are assigned to management VLAN 16.

Figure 128: Discovery of Newly Installed Switches



HSRP and Standby Cluster Command Switches

The switch supports Hot Standby Router Protocol (HSRP) so that you can configure a group of standby cluster command switches. Because a cluster command switch manages the forwarding of all communication and configuration information to all the cluster member switches, we strongly recommend the following:

- For a cluster command switch stack, a standby cluster command switch is necessary if the entire switch stack fails. However, if only the stack master in the command switch stack fails, the switch stack elects a new stack master and resumes its role as the cluster command switch stack.
- For a cluster command switch that is a standalone switch, configure a standby cluster command switch to take over if the primary cluster command switch fails.

A *cluster standby group* is a group of command-capable switches that meet the requirements described in the “Standby Cluster Command Switch Characteristics” section. Only one cluster standby group can be assigned per cluster.



Note The cluster standby group is an HSRP group. Disabling HSRP disables the cluster standby group.

The switches in the cluster standby group are ranked according to HSRP priorities. The switch with the highest priority in the group is the *active cluster command switch* (AC). The switch with the next highest priority is the *standby cluster command switch* (SC). The other switches in the cluster standby group are the *passive cluster command switches* (PC). If the active cluster command switch and the standby cluster command switch become disabled *at the same time*, the passive cluster command switch with the highest priority becomes the active cluster command switch. The HSRP **standby priority** interface configuration commands are the same for changing the priority of cluster standby group members and router-redundancy group members.



Note The HSRP standby hold time interval should be greater than or equal to three times the hello time interval. The default HSRP standby hold time interval is 10 seconds. The default HSRP standby hello time interval is 3 seconds.

Virtual IP Addresses

You need to assign a unique virtual IP address and group number and name to the cluster standby group. This information must be configured on a specific VLAN or routed port on the active cluster command switch. The active cluster command switch receives traffic destined for the virtual IP address. To manage the cluster, you must access the active cluster command switch through the virtual IP address, not through the command-switch IP address. This is in case the IP address of the active cluster command switch is different from the virtual IP address of the cluster standby group.

If the active cluster command switch fails, the standby cluster command switch assumes ownership of the virtual IP address and becomes the active cluster command switch. The passive switches in the cluster standby group compare their assigned priorities to decide the new standby cluster command switch. The passive standby switch with the highest priority then becomes the standby cluster command switch. When the previously active cluster command switch becomes active again, it resumes its role as the active cluster command switch, and the current active cluster command switch becomes the standby cluster command switch again. For more information about IP address in switch clusters, see the “IP Addresses” section.

Other Considerations for Cluster Standby Groups

These requirements also apply:

- Standby cluster command switches must be the same type of switches as the cluster command switch. For example, if the cluster command switch is a Catalyst 3750-E or Catalyst 3750-X switch, the standby cluster command switches must also be Catalyst 3750-E or Catalyst 3750-X switches. See the switch configuration guide of other cluster-capable switches for their requirements on standby cluster command switches.

If your switch cluster has a Catalyst 3750-X switch or a switch stack, it should be the cluster command switch. If not, when the cluster has a Catalyst 3750-E switch or switch stack, that switch should be the cluster command switch.

- Only one cluster standby group can be assigned to a cluster. You can have more than one router-redundancy standby group.

An HSRP group can be both a cluster standby group and a router-redundancy group. However, if a router-redundancy group becomes a cluster standby group, router redundancy becomes disabled on that group. You can re-enable it by using the CLI.

- All standby-group members must be members of the cluster.

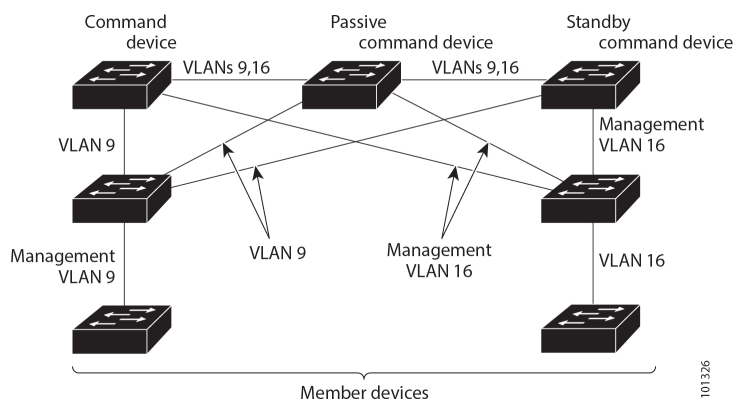


Note There is no limit to the number of switches that you can assign as standby cluster command switches. However, the total number of switches in the cluster—which would include the active cluster command switch, standby-group members, and cluster member switches—cannot be more than 16.

- Each standby-group member (Figure below) must be connected to the cluster command switch through the same VLAN. In this example, the cluster command switch and standby cluster command switches are Catalyst 3560-E, Catalyst 3750-E, Catalyst 3560-X, or Catalyst 3750-X cluster command switches. Each standby-group member must also be redundantly connected to each other through at least one VLAN in common with the switch cluster.

Catalyst 1900, Catalyst 2820, Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL cluster member switches must be connected to the cluster standby group through their management VLANs.

Figure 129: VLAN Connectivity between Standby-Group Members and Cluster Members



Automatic Recovery of Cluster Configuration

The active cluster command switch continually forwards cluster-configuration information (but not device-configuration information) to the standby cluster command switch. This ensures that the standby cluster command switch can take over the cluster immediately after the active cluster command switch fails.

Automatic discovery has these limitations:

- This limitation applies only to clusters that have Catalyst 2950, Catalyst 2960, Catalyst 2970, Catalyst 3550, Catalyst 3560, Catalyst 3560-E, Catalyst 3560-X, Catalyst 3750, Catalyst 3750-E, and Catalyst 3750-X command and standby cluster command switches: If the active cluster command switch and standby cluster command switch become disabled *at the same time*, the passive cluster command switch with the highest priority becomes the active cluster command switch. However, because it was a passive standby cluster command switch, the previous cluster command switch *did not* forward cluster-configuration information to it. The active cluster command switch only forwards cluster-configuration information to the standby cluster command switch. You must therefore rebuild the cluster.
- This limitation applies to all clusters: If the active cluster command switch fails and there are more than two switches in the cluster standby group, the new cluster command switch does not discover any Catalyst 1900, Catalyst 2820, and Catalyst 2916M XL cluster member switches. You must re-add these cluster member switches to the cluster.
- This limitation applies to all clusters: If the active cluster command switch fails and becomes active again, it does not discover any Catalyst 1900, Catalyst 2820, and Catalyst 2916M XL cluster member switches. You must again add these cluster member switches to the cluster.

When the previously active cluster command switch resumes its active role, it receives a copy of the latest cluster configuration from the active cluster command switch, including members that were added while it was down. The active cluster command switch sends a copy of the cluster configuration to the cluster standby group.

IP Addresses

You must assign IP information to a cluster command switch. You can assign more than one IP address to the cluster command switch, and you can access the cluster through any of the command-switch IP addresses. If you configure a cluster standby group, you must use the standby-group virtual IP address to manage the cluster from the active cluster command switch. Using the virtual IP address ensures that you retain connectivity to the cluster if the active cluster command switch fails and that a standby cluster command switch becomes the active cluster command switch.

If the active cluster command switch fails and the standby cluster command switch takes over, you must either use the standby-group virtual IP address or any of the IP addresses available on the new active cluster command switch to access the cluster.

You can assign an IP address to a cluster-capable switch, but it is not necessary. A cluster member switch is managed and communicates with other cluster member switches through the command-switch IP address. If the cluster member switch leaves the cluster and it does not have its own IP address, you must assign an IP address to manage it as a standalone switch.

Hostnames

You do not need to assign a host name to either a cluster command switch or an eligible cluster member. However, a hostname assigned to the cluster command switch can help to identify the switch cluster. The default hostname for the switch is *Switch*.

If a switch joins a cluster and it does not have a hostname, the cluster command switch appends a unique member number to its own hostname and assigns it sequentially as each switch joins the cluster. The number means the order in which the switch was added to the cluster. For example, a cluster command switch named *eng-cluster* could name the fifth cluster member *eng-cluster-5*.

If a switch has a hostname, it retains that name when it joins a cluster and when it leaves the cluster.

If a switch received its hostname from the cluster command switch, was removed from a cluster, was then added to a new cluster, and kept the same member number (such as 5), the switch overwrites the old hostname (such as *eng-cluster-5*) with the hostname of the cluster command switch in the new cluster (such as *mkg-cluster-5*). If the switch member number changes in the new cluster (such as 3), the switch retains the previous name (*eng-cluster-5*).

Passwords

You do not need to assign passwords to an individual switch if it will be a cluster member. When a switch joins a cluster, it inherits the command-switch password and retains it when it leaves the cluster. If no command-switch password is configured, the cluster member switch inherits a null password. Cluster member switches only inherit the command-switch password.

If you change the member-switch password to be different from the command-switch password and save the change, the switch is not manageable by the cluster command switch until you change the member-switch password to match the command-switch password. Rebooting the member switch does not revert the password back to the command-switch password. We recommend that you do not change the member-switch password after it joins a cluster.

For password considerations specific to the Catalyst 1900 and Catalyst 2820 switches, see the installation and configuration guides for those switches.

SNMP Community Strings

A cluster member switch inherits the command-switch first read-only (RO) and read-write (RW) community strings with *@esN* appended to the community strings:

- *command-switch-readonly-community-string@esN*, where N is the member-switch number.
- *command-switch-readwrite-community-string@esN*, where N is the member-switch number.

If the cluster command switch has multiple read-only or read-write community strings, only the first read-only and read-write strings are propagated to the cluster member switch.

The switches support an unlimited number of community strings and string lengths.

For SNMP considerations specific to the Catalyst 1900 and Catalyst 2820 switches, see the installation and configuration guides specific to those switches.

TACACS+ and RADIUS

If Terminal Access Controller Access Control System Plus (TACACS+) is configured on a cluster member, it must be configured on all cluster members. Similarly, if RADIUS is configured on a cluster member, it must be configured on all cluster members. Further, the same switch cluster cannot have some members configured with TACACS+ and other members configured with RADIUS.

LRE Profiles

A configuration conflict occurs if a switch cluster has Long-Reach Ethernet (LRE) switches that use both private and public profiles. If one LRE switch in a cluster is assigned a public profile, all LRE switches in that cluster must have that same public profile. Before you add an LRE switch to a cluster, make sure that you assign it the same public profile used by other LRE switches in the cluster.

A cluster can have a mix of LRE switches that use different private profiles.

Using the CLI to Manage Switch Clusters

You can configure cluster member switches from the CLI by first logging into the cluster command switch. Enter the **rcommand** user EXEC command and the cluster member switch number to start a Telnet session (through a console or Telnet connection) and to access the cluster member switch CLI. The command mode changes, and the Cisco IOS commands operate as usual. Enter the **exit** privileged EXEC command on the cluster member switch to return to the command-switch CLI.

This example shows how to log into member-switch 3 from the command-switch CLI:

```
switch# rcommand 3
```

If you do not know the member-switch number, enter the **show cluster members** privileged EXEC command on the cluster command switch. For more information about the **rcommand** command and all other cluster commands, see the switch command reference.

The Telnet session accesses the member-switch CLI at the same privilege level as on the cluster command switch. The Cisco IOS commands then operate as usual.



Note The CLI supports creating and maintaining switch clusters with up to 16 switch stacks.

Using SNMP to Manage Switch Clusters

When you first power on the switch, SNMP is enabled if you enter the IP information by using the setup program and accept its proposed configuration. If you did not use the setup program to enter the IP information and SNMP was not enabled, you can enable it as described in the “Configuring SNMP”. On Catalyst 1900 and Catalyst 2820 switches, SNMP is enabled by default.

When you create a cluster, the cluster command switch manages the exchange of messages between cluster member switches and an SNMP application. The cluster software on the cluster command switch appends the cluster member switch number (*@esN*, where N is the switch number) to the first configured read-write and read-only community strings on the cluster command switch and propagates them to the cluster member switch. The cluster command switch uses this community string to control the forwarding of gets, sets, and get-next messages between the SNMP management station and the cluster member switches.

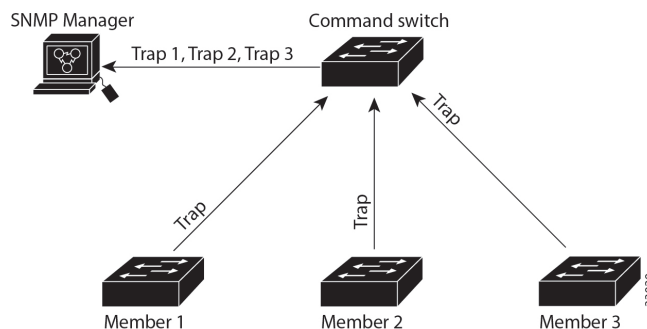


Note When a cluster standby group is configured, the cluster command switch can change without your knowledge. Use the first read-write and read-only community strings to communicate with the cluster command switch if there is a cluster standby group configured for the cluster.

If the cluster member switch does not have an IP address, the cluster command switch redirects traps from the cluster member switch to the management station, as shown in the Figure. If a cluster member switch has its own IP address and community strings, the cluster member switch can send traps directly to the management station, without going through the cluster command switch.

If a cluster member switch has its own IP address and community strings, they can be used in addition to the access provided by the cluster command switch.

Figure 130: SNMP Management for a Cluster





CHAPTER 80

Configuring AVC with DNS-AS

- [Prerequisites for AVC with DNS-AS, on page 1639](#)
- [Restrictions and Guidelines for AVC with DNS-AS, on page 1639](#)
- [Information About AVC with DNS-AS, on page 1640](#)
- [How to Configure AVC with DNS-AS, on page 1644](#)
- [Monitoring AVC with DNS-AS, on page 1657](#)
- [Troubleshooting AVC with DNS-AS, on page 1660](#)
- [Feature History and Information for AVC with DNS-AS, on page 1661](#)

Prerequisites for AVC with DNS-AS

- You have the [Cisco ONE for Access](#) license to use AVC with DNS-AS.
- You have enabled Multilayer Switch (MLS) Quality of Service (QoS).
- You have maintained metadata in the authoritative DNS server and reachability exists - before you enable AVC with DNS-AS
- The DNS-AS client can snoop forward look-up requests originating from hosts.
- To ensure DNS packet logging or snooping, you have attached the policy map to the interface, by using the **service-policy input** command.

Restrictions and Guidelines for AVC with DNS-AS

- The feature is supported only on the Cisco Catalyst 3560-CX Series Switches and not the Cisco Catalyst 2960-CX Series Switches
- Only a forward look-up is supported.
- Two DNS servers are supported (in case of a failover). One is considered the primary DNS server and other, the secondary DNS server.
- IPv6 is not supported—AAAA requests, and IPv6 DNS servers are not supported.
- AVC with DNS-AS is supported only on physical interfaces, in the ingress direction.
- Virtual Routing and Forwarding (VRF) is not supported.

- We recommend a maximum of 300 AVC with DNS-AS applications (domain names) in the binding table, because of its effect on the ternary content addressable memory (TCAM). To know how the addition of applications affects the TCAM see the [Troubleshooting AVC with DNS-AS](#) section of this chapter.

Information About AVC with DNS-AS

The Application Visibility Control (AVC) with Domain Name System as an Authoritative Source (DNS-AS) feature (AVC with DNS-AS) provides a centralized means of controlling the identification and classification of trusted network traffic in an organization. It accomplishes this by using network metadata stored in a DNS server that is authoritative to the domain in question, to identify applications, Quality of Service (QoS) to classify the corresponding traffic and apply suitable policies, and Flexible NetFlow (FNF), to monitor and export application information to an external collector.

The feature provides:

- Application Visibility—Ensuring unambiguous visibility of applications.

The DNS-AS mechanism snoops requests and does not require a CPU-intensive, deep packet inspection (DPI). Since traffic classification is by means of a DNS request and not DPI, this feature is compatible in scenarios where network traffic is encrypted.

- Metadata Driven—Using information about applications.

You can program the network holistically so it behaves like a self-driving car. You now have information about all the required applications in your network, irrespective of whether traffic is encrypted or not.

- Centralized Control—Using a cross-domain application intent policy controller.

The feature leverages an existing, universally available query-response mechanism to enable local DNS servers within an organization to act as authoritative servers and propagate application classification information to DNS-AS clients (switches) in an enterprise network.

- Control without Administrative Access—Proving alternatives to controller-based approaches.

The feature supports scenarios where your network may be in the cloud and you may not own it. You can still control network devices across the Internet, even though you may not have administrative control of these devices.

Overview of AVC with DNS-AS

The process starts with an organization's requirements relating to management and control of network traffic. You begin by assessing the software applications that run on the various hosts (phones, PCs etc.) in your network, the domains (websites) and applications accessed by these devices, and the business-relevance of these domains and applications in your organization.

The assessment helps you arrive at a list of domains and applications that are "trusted" by your organization, designating all remaining domains and applications as untrusted.

With DNS-AS enabled on your network and the list of trusted domains at hand, the networking devices or DNS-AS clients in your network identify which applications the network traffic belongs to or which domains are being requested. As long as the traffic is part of the trusted list, the switch requests the DNS server for metadata and IP address information. This request is sent in the form of a DNS-query. The response, once

received, is cached locally until the Time-to-Live (TTL) for that resource record expires. The response is bound to the traffic and allows the DNS-AS client to now identify, classify, and forward traffic accordingly.

Key Concepts for AVC with DNS-AS

Concept	Meaning or Definition
Metadata (RFC6759)	<p>In the context of the AVC with DNS-AS feature, this includes traffic classification information, application identification information, and business relevance information.</p> <p>Metadata is maintained in the form of TXT records. The following is a sample metadata record in the prescribed format:</p> <pre>CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202</pre>
Forward look-up	<p>A request for an IP address or a request for an “A” record, originating from a host.</p> <p>Being able to snoop these forward lookups in the network traffic is fundamental to the AVC with DNS-AS feature.</p>
Host	<p>A PC or mobile where users run software applications, access websites and so on.</p> <p>Forward look-up requests originate from hosts.</p>
Client or DNS-AS client	<p>Networking devices throughout your network. Host traffic is always routed through such a client.</p> <p>Note This chapter deals with the configuration of the AVC with DNS-AS on Cisco Catalyst Switches that are deployed as access switches only. Throughout this document, the term client, DNS-AS client, refers to the switch where AVC with DNS-AS is enabled.</p> <p>DNS-AS clients receive metadata from an authoritative DNS server and maintain a database of this information in the form of records. How long the record remains in the client’s database, is determined by the record’s TTL.</p>
Binding table	<p>A table that resides in the DNS-AS client and serves as a database of parsed DNS server responses [TXT records and “A” records].</p> <p>Every DNS-AS client has a binding table of its own.</p> <p>This table not to be confused with the trusted domain list which is only a list of the trusted domains.</p>
"A" record	<p>A record containing the domain name and IP address information [Only IPv4 address]. This is one of the DNS-Server responses (the other being the TXT record) and has a predefined lifespan.</p> <p>A forward lookup request from a host is a request for an “A” record.</p>

Concept	Meaning or Definition
TXT DNS-AS resource record or TXT record	<p>A record containing metadata. This is one of the DNS-Server responses (the other being the “A” record) and has a predefined lifespan.</p> <p>A TXT record is limited to 255 characters.</p> <p>For AVC with DNS-AS, the TXT attribute is always CISCO-CLS. Any TXT record that starts with CISCO-CLS= can be recognized as an AVC with DNS-AS message. The message format is as follows:</p> <pre>CISCO-CLS=<option>:<val>{ <option>:<val>}*</pre>
Time-to-Live (TTL)	<p>The lifespan of an “A” record and TXT record in the binding table.</p> <p>TTL values are configured on the DNS server.</p> <p>While a TTL accompanies both TXT and “A” record responses, the DNS client only goes by the “A” record response from the DNS server.</p>
Authoritative DNS server	<p>The go-to DNS server for all client metadata and “A” record requests.</p> <p>Every DNS domain has only one authoritative DNS server.</p> <p>Such a server maintains records of application metadata in the form of a TXT record, and only returns responses to queries about domain names that have been maintained in the required format.</p> <p>The following is a sample metadata record in the prescribed format:</p> <pre>CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202</pre>

AVC with DNS-AS Process Flow

The working of AVC with DNS-AS involves the DNS snooping process and the DNS-AS client process—both of which are loosely coupled, but independent processes.

DNS Snooping Process

-
- Step 1** The host initiates an “A” record request.
- A user from your organization is in a meeting room in an office building. The associated DNS-AS client here is a switch (Network traffic from this meeting room is routed through this switch). The user looks up a website `www.example.com`, which initiates the request for an “A” record.
- Step 2** The authoritative DNS-server responds with an “A” record response.
-

DNS-AS Client Process

-
- Step 1** The DNS-AS client sends a DNS query (TXT request) to the authoritative DNS server.

The DNS-AS client, which is constantly snooping for requests (that correspond with entries in the trusted domain list), finds the host's forward look-up request. Based on the snooped result, the DNS-AS client sends a TXT request to the authoritative DNS server.

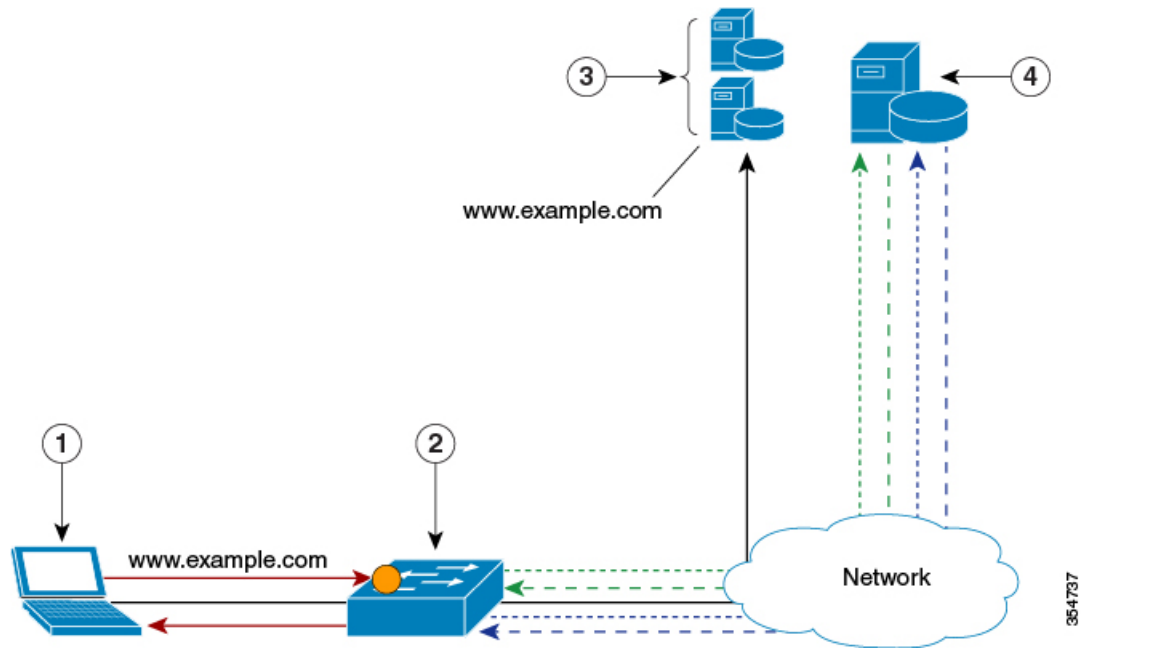
Note The DNS-AS client receives a copy of the host's "A" record request, and does not alter the host's original request in any manner.

- Step 2** The authoritative DNS-server responds with a TXT record response.
- Step 3** A successful TXT response is followed by an "A" record request.
- Step 4** The authoritative DNS-server responds with an "A" record response.
- Step 5** The DNS-AS client parses and saves the response in its binding table.




The DNS-AS client saves the TXT record and "A" record in its binding table. The response will remain saved in the binding table for the duration specified by the TTL of the "A" record. The system automatically checks and prevents duplicate entries for a fully qualified domain name in the binding table.

The DNS-AS client uses the metadata it receives (from the DNS Server), to determine if a QoS policy should be applied. The DNS-AS client forwards information about identified applications, to FNF, enabling you to export this information.

Figure: AVC with DNS-AS Process Flow



1	Host	2	DNS-As Client	3	Authoritative DNS Server
Part I: DNS Snooping Process					
----->	An "A" record request from the host to the DNS server	-----<	An "A" record response from the DNS server to the host		

Part II: DNS-As Client Process			
	A copy of the host's "A" record request that the DNS-AS client saves	-	-
	TXT record and "A" record request from the DNS-AS client to the DNS server		TXT record and "A" record response from the DNS server to the DNS-AS client

Default Configuration for AVC with DNS-AS

DNS-AS is disabled.

How to Configure AVC with DNS-AS

Generating Metadata Streams

Application metadata is configured and saved on the local, authoritative DNS server. You configure application classification information, for each trusted domain, in a prescribed format (a metadata stream). This is the information that the server propagates to switches when queried for application metadata. When the switch sends a TXT query regarding an application, the DNS server sends the relevant metadata in the TXT response.

To generate metadata streams, perform the following task:

SUMMARY STEPS

1. Go to the: [AVC Resource Record Generator](#).
2. Click one of the options to generate the metadata stream.
 - Generate predefined
 - Generate custom
3. Copy metadata into the corresponding TXT Resource Record of the DNS server in charge of the DNS domain that you have marked as a trusted domain.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Go to the: AVC Resource Record Generator . Example: <code>CTSD0-CLS=app-name:example app-class:TD business:YES app-id:CU/28202</code>	Helps you generate a metadata stream for an application or domain, in a TXT record format. You can specify the following metadata fields: <ul style="list-style-type: none"> • (Optional) Domain Name • (Mandatory) Application Name—A value is mandatory. This can be an existing application name or custom application name.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Existing Application Name (app-name:)—Select from the list of standard applications. • (Optional) Custom Application Name(app-name:)—If you enter a custom application name, you must also maintain the Traffic Class and Business Relevance information in the metadata stream. • (Optional) Selector ID (app-id:)—Consists of a classification engine ID (first eight bits) and a selector ID (the next twenty-four bits). <ul style="list-style-type: none"> • Engine ID or Classification Engine ID—Defines the context for the selector ID. Only these engine IDs are allowed: <ul style="list-style-type: none"> L3—IANA layer 3 protocol number L4—IANA layer 4 well-known port number L7—Cisco global application ID CU—Custom protocol. Use this engine ID for custom application names. • Selector ID—An application identifier, for a given classification engine ID. Enter a numeric value between 1 and 65535 <p>Note When you enter the engine ID and selector ID for existing application names, be sure to align with the Network Based Application Recognition (NBAR) standard. Only then will the FNF exporters report with a common ID and in a consistent manner.</p> <ul style="list-style-type: none"> • (Optional) Port Range (server-port:) • (Optional) Traffic Class (app-class:) • (Optional) Business Relevance (business:)—If you do not select yes or no, the business relevance value is set based on the app-class or app-name, in that order of priority. <p>For information about how traffic class and business relevance fields here map to QoS traffic classification, see App-Class and QoS Traffic Mapping</p>

	Command or Action	Purpose
Step 2	Click one of the options to generate the metadata stream. <ul style="list-style-type: none"> • Generate predefined • Generate custom Example: Generate predefined	Generate predefined —Generate a predefined metadata stream for well known applications, using best practice defaults. Generate custom —Generate a custom metadata stream for your own applications using custom values.
Step 3	Copy metadata into the corresponding TXT Resource Record of the DNS server in charge of the DNS domain that you have marked as a trusted domain.	Copy and paste the metadata stream from the website, to the authoritative DNS server you are using.

Configuring a DNS Server as the Authoritative Server

All DNS-AS clients in the network should be configured to send all DNS queries to one authoritative DNS server. On a Cisco Catalyst switch, perform the following task:

SUMMARY STEPS

1. **configure terminal**
2. **ip name-server***server-address*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	ip name-server <i>server-address</i> Example: Switch(config)# ip name-server server-address 192.0.2.1 192.0.2.2	Specifies the address of the authoritative DNS server. The port number is always 53. You can configure up to two DNS Servers, in case of a failover. Note The command allows you configure up to six name servers (IPv4 and IPv6). Ensure that at least the first two IP addresses in the sequence are IPv4 addresses, because the AVC with DNS-AS feature will use only these. See the example below, here the first two addresses are IPv4 (192.0.2.1 and 192.0.2.2), the third one (2001:DB8::1) is an IPv6 address. AVC with DNS-AS will use the first two. Switch(config)# ip name-server 192.0.2.1 192.0.2.2 2001:DB8::1

Enabling AVC with DNS-AS

DNS-AS is disabled by default. To enable the feature on a Cisco Catalyst switch, perform the following task:

SUMMARY STEPS

1. **configure terminal**
2. **[no] avc dns-as client enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	[no] avc dns-as client enable Example: <pre>Switch(config)# avc dns-as client enable</pre>	Enables AVC with DNS-AS on the switch (DNS-AS client). The system then creates a binding table where parsed DNS server responses are stored till the TTL expires. Note To ensure DNS packet logging or snooping, you must attach the policy map (containing the relevant class maps that will determine traffic class) to the interface by using the service-policy input command. For more information, see Configuring QoS for AVC with DNS-AS, on page 1648

Maintaining the List of Trusted Domains

Trusted domains are saved in every DNS-AS client where AVC with DNS-AS is enabled. When the feature is first enabled on the DNS-AS client, the list is empty. You must enter the domains that the switch should trust. The switch snoops only for network traffic that is maintained in this list. To make entries in the trusted domain list, perform the following task:

SUMMARY STEPS

1. **configure terminal**
2. **[no] avc dns-as client trusted-domains**
3. **[no] domain *domain-name***

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Switch# <code>configure terminal</code>	
Step 2	<p>[no] <code>avc dns-as client trusted-domains</code></p> <p>Example:</p> <pre>Switch(config)# <code>avc dns-as client trusted-domains</code></pre>	Enters the trusted domain configuration mode.
Step 3	<p>[no] <code>domain domain-name</code></p> <p>Example:</p> <pre>Switch(config-trusted-domains)# <code>domain www.example.com</code> OR Switch(config-trusted-domains)# <code>domain *example.com</code></pre>	<p>Enter the domain name you would like to add to the trusted domain list. This forms part of the list of trusted domains for the DNS-AS client. All remaining domains are ignored and will follow default forwarding behavior.</p> <p>You can enter up to 50 domains.</p> <p>You can use regular expressions to match the domain name. For example, to represent all the domains for an organization, if you enter: <code>Switch(config-trusted-domains)# domain *.example.*</code>, the DNS-AS client matches <code>www.example.com</code>, <code>ftp.example.org</code> and any other domain that pertains to the organization “example”. But use such an entry at your discretion, because it could increase the size of the binding table considerably.</p>

Configuring QoS for AVC with DNS-AS

In order to isolate and classify trusted traffic as defined in the metadata stream, you must create class maps (one for each traffic class) > define traffic-class match criteria and business-relevance match criteria > create a policy map > add the class map > set action > attach the policy map to the interface. For more information, see the [Classification Overview](#) section of the *Configuring QoS* chapter in this guide.

Class Map Configuration in the Easy QoS Model

In order to determine the number of traffic classes that should be provisioned, you can use the 12-class Easy QoS Model. This model provides a uniform, standards-based recommendations to help ensure that QoS designs and deployments are unified and consistent across an organization. The following sample output displays class map configuration for traffic class and business relevance, according to the 12-class Easy QoS Model:



Note Only in the context of the DNS-AS feature, you can specify up to two match attributes for each class.

```
class-map match-all VOICE
match protocol attribute traffic-class voip-telephony
match protocol attribute business-relevance business-relevant
class-map match-all BROADCAST-VIDEO
match protocol attribute traffic-class broadcast-video
match protocol attribute business-relevance business-relevant
```



```

class-map match-all REAL-TIME-INTERACTIVE
match protocol attribute traffic-class real-time-interactive
match protocol attribute business-relevance business-relevant
class-map match-all MULTIMEDIA-CONFERENCING
match protocol attribute traffic-class multimedia-conferencing
match protocol attribute business-relevance business-relevant
class-map match-all MULTIMEDIA-STREAMING
match protocol attribute traffic-class multimedia-streaming
match protocol attribute business-relevance business-relevant
class-map match-all SIGNALING
match protocol attribute traffic-class signaling
match protocol attribute business-relevance business-relevant
class-map match-all NETWORK-CONTROL
match protocol attribute traffic-class network-control
match protocol attribute business-relevance business-relevant
class-map match-all NETWORK-MANAGEMENT
match protocol attribute traffic-class ops-admin-mgmt
match protocol attribute business-relevance business-relevant
class-map match-all TRANSACTIONAL-DATA
match protocol attribute traffic-class transactional-data
match protocol attribute business-relevance business-relevant
class-map match-all BULK-DATA
match protocol attribute traffic-class bulk-data
match protocol attribute business-relevance business-relevant
class-map match-all SCAVENGER
match protocol attribute business-relevance business-irrelevant

```

Policy Map Definitions in the Easy QoS Model

The following sample output displays the policy map definitions, with traffic attribute marking for all the traffic classes in the 12-class Easy QoS Model:

```

policy-map MARKING
class VOICE
set dscp ef
class BROADCAST-VIDEO
set dscp cs5
class REAL-TIME-INTERACTIVE
set dscp cs4
class MULTIMEDIA-CONFERENCING
set dscp af41
class MULTIMEDIA-STREAMING
set dscp af31
class SIGNALING
set dscp cs3
class NETWORK-CONTROL
set dscp cs6
class NETWORK-MANAGEMENT
set dscp cs2
class TRANSACTIONAL-DATA
set dscp af21
class BULK-DATA
set dscp af11
class SCAVENGER
set dscp cs1
class class-default
set dscp default

```

App-Class and QoS Traffic Mapping

The following table shows how the `app-class` field in the metadata stream maps to the 12-class Easy QoS Model of traffic classification.

App-Class and QoS Traffic Mapping

Application Class Long Text	Application Class Short Text	Corresponding QoS Traffic Class Name and Business Relevance
VOIP-TELEPHONY	VO	Traffic-class = voip-telephony Business-relevance = YES
BROADCAST-VIDEO	BV	Traffic-class = broadcast-video Business-relevance = YES
REALTIME-INTERACTIVE	RTI	Traffic-class = real-time-interactive Business-relevance = YES
MULTIMEDIA-CONFERENCING	MMC	Traffic-class = multimedia-conferencing Business-relevance = YES
MULTIMEDIA-STREAMING	MMS	Traffic-class = multimedia-streaming Business-relevance = YES
NETWORK-CONTROL	NC	Traffic-class = network-control Business-relevance = YES
SIGNALING	CS	Traffic-class = Signaling Business-relevance = YES
OPS-ADMIN-MGMT	OAM	Traffic-class = ops-admin-mgmt Business-relevance = YES
TRANSACTIONAL-DATA	TD	Traffic-class = Transactional-Data Business-relevance = YES
BULK-DATA	BD	Traffic-class = bulk-data Business-relevance = YES
BEST-EFFORT	BE	Traffic-class = <no change> Business-relevance = default
SCAVENGER	SCV	Traffic-Class = <no change> Business-relevance = NO

Classifying Network Control Traffic

The following example shows how to classify network control traffic. The corresponding metadata that should be maintained is: `CISCO-CLS=app-name:example|app-class:NC|business:YES`

1. Create class maps and match attributes:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# class-map NETWORK-CONTROL
Switch(config-cmap)# match protocol attribute traffic-class network-control
Switch(config-cmap)# match protocol attribute business-relevance business-relevant
Switch(config-cmap)# end
```

2. Create the policy map, attach the class map to it and specify priority:

```
Switch# configure terminal
Switch configuration commands, one per line. End with CNTL/Z.
Switch(config)# policy-map MARKING
Switch(config-pmap)# class NETWORK-CONTROL
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# end
```

3. Attach the policy map to an interface:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface tengigabitethernet 1/0/1
Switch(config-if)# service-policy input MARKING
Switch(config-if)# end
```

Configuring FNF for AVC with DNS-AS

With FNF you can gain visibility into the applications running on your network, and use FNF option templates to export application ID, description, and attribute information. You must configure these FNF settings on the DNS-AS client:

- Configure a flow record to collect nonkey field **application-name**, and the key fields **ipv4 source address** and **ipv4 destination address**
- Configure a flow exporter and the two option templates. Option templates fetch application information.
 - Option template **application-table**, exports only applications resolved by the DNS-AS client, that is, the application ID and name from the binding table. The corresponding application descriptions are from Network Based Application Recognition (NBAR) definition for standard applications. A constructed help string is used for custom applications..
 - Option template **application-attributes** fetches attribute information by mapping it to the application name. Where standard application names are used, the option template uses standard Network Based Application Recognition (NBAR) attribute definitions; where custom application names are used, user-defined application names and only certain attribute fields are guaranteed to carry values.
- Configure a flow monitor and apply it to an interface to enable network traffic monitoring.

FNF Interaction with DNS-AS—With every flow that is created in the flow table, the DNS-AS client resolves the application name for the flow (if the entry exists in the binding table), by using the destination IP address (and if not available), the source IP address.

At periodic, configured intervals (600 seconds, by default), FNF exports option template data, that is mapped to the corresponding application name, to an external collector.

Option Templates

The **application-table** and **application-attributes** option templates are supported. Option templates determine the information that is exported to an external collector.

option application-table

This template exports the application name, application tag, and description to the external collector.

On a device where AVC with DNS-AS is enabled, only applications resolved by the DNS-AS client are exported. But as a permanent feature, the application-table template exports applications **unclassified** and **unknown**, irrespective of whether the feature is enabled or not.

- Application Name—For custom and standard applications, this information is derived from the TXT response (**app-name:**) that is saved in the binding table.
- Application Tag—The same as the application ID in the context of the AVC with DNS-AS feature. It consists of the engine ID and selector ID.
 - Engine ID or Classification Engine ID—Defines the context for the selector ID. Only these values are supported:
 - L3—IANA layer 3 protocol number (IANA_L3_STANDARD, ID: 1)
 - L4—IANA layer 4 well-known port number (IANA_L4_STANDARD, ID: 3)
 - L7—Cisco global application ID (CISCO_L7_GLOBAL, ID: 13)
 - CU—Custom protocol, (NBAR_CUSTOM, ID: 6)
 - Selector ID—Uniquely identifies the application or classification.

For standard applications, the application tag information is derived from these sources, in the given order of precedence:

1. TXT response (**app-id:**)
2. The NBAR definition for standard applications (if the TXT response does not carry a value).

For custom applications, the following applies to application tag information:

- It is derived only from the TXT response (**app-id:**)
- For the engine ID, the DNS-AS client automatically uses CU—Custom protocol, (NBAR_CUSTOM, ID: 6).
- For the selector ID, the DNS-AS client allots a custom selector ID. A maximum of 120 custom applications are supported - out of which 110 are available to the DNS-AS client. Starting with selector ID value 243, IDs are assigned in descending order. When there are no remaining IDs to assign, the entry is not saved in the binding table.
- Description—This information is derived from the NBAR definition for standard applications. For custom applications, the DNS-AS client uses: User Defined Protocol <app-name>.

option application-attributes

This template enables the collector to map the application names (from the option application-table), to attributes. Attributes are statically assigned to each protocol or application, and are not dependent on traffic. The template supports the following attributes:

For standard applications—

- Application Tag—See the Application Tag info in the [option application-table](#) section above. The same applies here.
- Category—Groups applications based on the first level of categorization for each protocol as the match criteria. Similar applications are grouped together under one category. For example, the email category contains all email applications such as, Internet Mail Access Protocol (IMAP), Simple Mail Transfer Protocol (SMTP), Lotus Notes, and so on.
- Sub-category—Groups applications based on the second level of categorization for each protocol as the match criteria. For example, clearcase, dbase, rda, mysql and other database applications are grouped under the database group.
- Application Group—Groups the same networking applications together. For instance, Example-Messenger, Example-VoIP-messenger, and Example-VoIP-over-SIP are grouped together under the example-messenger-group
- Peer-to-peer (p2p)—Groups protocols based on whether or not they use p2p technology.
- Tunnel—Groups protocols based on whether or not a protocol tunnels the traffic of other protocols. Protocols for which the NBAR does not provide any value are categorized under the unassigned tunnel group. For example, Layer 2 Tunneling Protocols (L2TP).
- Encryption—Groups applications based on the encrypted and nonencrypted status of the applications. Protocols for which the NBAR does not provide any value are categorized under the unassigned encrypted group.
- Traffic class—Groups applications and protocols based on the traffic class they belong to. For example, all applications that have traffic class TD. Traffic class information is derived from these sources, in the given order of precedence:
 1. TXT response (**app-class:**)
 2. The NBAR definition for standard applications (if the TXT response does not carry a value)
- Business relevance—Groups applications based on whether or not they have been marked as business-relevant. For example, all applications that have business relevance as YES. Business relevance information is derived from these sources, in the given order of precedence:
 1. TXT response (**business:**)
 2. The NBAR definition for standard applications (if the TXT response does not carry a value)

For custom applications—

Only these attributes of the application-attributes options template are guaranteed to carry a value:

- Application Tag—See the Application Tag info in the [option application-table](#) section above. The same applies here.
- Traffic class—This information is derived from the TXT response (**app-class:**)

- Business relevance—This information is derived from the TXT response (**business:**)

Sample FNF Configuration for AVC with DNS-AS

The following example shows how you can configure FNF for AVC with DNS-AS:

Part 1: Create a flow record. As in the example, you must configure:

- The source and destination IP addresses as key fields, in order to resolve application names.
- The use of the application name as a monkey field in flow record.

Additionally (not mandatory), you can also configure the number of bytes or packets in a flow as a monkey field, to display the number of applications sent to the collector

```
Switch# configure terminal
Switch(config)# flow record example-record1
Switch(config-flow-record)# match ipv4 source address
Switch(config-flow-record)# match ipv4 destination address
Switch(config-flow-record)# collect application name
Switch(config-flow-record)# collect counter packets
Switch(config-flow-record)# exit
```

```
Switch# show flow record example-record1
flow record example-record1
 match ipv4 source address
 match ipv4 destination address
 collect application name
 collect counter packets
```

Part 2: Create a flow exporter.

Also configure the **application-table** and **application-attributes** option templates in the exporter. Without option templates, the collector cannot retrieve meaningful application information. At a minimum we recommend that you configure the application-table option. For attribute information, also configure the application-attribute option.

You can also change the frequency of template export in seconds (the allowed range is 1 to 86400 seconds; the default is 600 seconds).

```
Switch(config)# flow exporter example-exporter1
Switch(config-flow-exporter)# option application-table
Switch(config-flow-exporter)# option application-attributes
Switch(config-flow-exporter)# template data timeout 500
Switch(config-flow-exporter)# exit
```

```
Switch# show flow exporter example-exporter1
Flow Exporter example-exporter1:
  Description:           User defined
  Export protocol:       NetFlow Version 9
  Transport Configuration:
    Destination IP address: 192.0.1.254
    Source IP address:     192.51.100.2
    Transport Protocol:    UDP
    Destination Port:      9995
    Source Port:           54964
    DSCP:                  0x0
    TTL:                   255
    Output Features:       Not Used
  Options Configuration:
    application-table (timeout 500 seconds)
```

```

application-attributes (timeout 500 seconds)

Switch# show flow exporter example-exporter1 statistics
Flow Exporter example-exporter1:
  Packet send statistics (last cleared 00:00:48 ago):
    Successfully sent:          2                (924 bytes)

  Client send statistics:
    Client: Option options application-name
      Records added:           4
      - sent:                   4
      Bytes added:             332
      - sent:                   332

    Client: Option options application-attributes
      Records added:           2
      - sent:                   2
      Bytes added:             388
      - sent:                   388

```

Part 3: Create a flow monitor

Apply the flow monitor to an interface, to perform network traffic monitoring.

You can also apply a QoS policy to the same interface. This example applies the QoS policy created as part of the sample QoS configuration ([Classifying Network Control Traffic, on page 1651](#))

```

Switch# configure terminal
Switch(config)# flow monitor example-monitor1
Switch(config-flow-monitor)# record example-record1
Switch(config-flow-monitor)# exporter exporter-exporter1
Switch(config-flow-monitor)# exit
Switch(config)# interface tengigabitethernet 1/0/1
Switch(config-if)# switchport access vlan 100
Switch(config-if)# switchport mode access
Switch(config-if)# ip flow monitor example-monitor1 input
Switch(config-if)# service-policy input MARKING
Switch(config-if)# end

Switch# show flow monitor
flow monitor example-monitor1
record example-record1
exporter example-exporter1
!
Switch# show interface tengigabitethernet1/0/1
interface tengigabitethernet1/0/1
switchport access vlan 100
switchport mode access
ip flow monitor example-monitor1 input

Switch# show flow monitor example-monitor1 cache
Cache type:                    Normal
Cache size:                    16640
Current entries:                3
High Watermark:                3

Flows added:                   6
Flows aged:                    3
- Active timeout ( 1800 secs)  0
- Inactive timeout ( 30 secs)  3
- Event aged                   0
- Watermark aged               0
- Emergency aged               0

```

```

IPV4 SOURCE ADDRESS:      192.0.1.254
IPV4 DESTINATION ADDRESS: 192.51.100.2
counter packets long:    7479
application name:        appexample1

IPV4 SOURCE ADDRESS:      192.51.100.11
IPV4 DESTINATION ADDRESS: 203.0.113.125
counter packets long:    445
application name:        appexample2

IPV4 SOURCE ADDRESS:      192.51.51.51
IPV4 DESTINATION ADDRESS: 203.0.113.100
counter packets long:    14325
application name:        appexample3
Switch#

```

Part 4: Other related **show** commands

```

Switch# show avc dns-as client binding-table detail
DNS-AS generated protocols:
Max number of protocols      :50
Customization interval [min] :N/A

Age          : The amount of time that the entry is active
TTL          : Time to live which was learned from DNS-AS server
Time To Expire : Entry expiration time in case device does not see DNS traffic for the
entry host

Protocol-Name      : appexample1
VRF                : <default>
Host               : www.appexample1.com
Age[min]           : 2
TTL[min]           : 60
Time To Expire[min] : 58
TXT Record         : app-name:appexample1|app-class:VO|business:YES
Traffic Class      : voip-telephony
Business Relevance : business relevant
IP                 : 192.0.1.254

Protocol-Name      : appexample2
VRF                : <default>
Host               : www.appexample2.com
Age[min]           : 2
TTL[min]           : 60
Time To Expire[min] : 58
TXT Record         : app-name:appexample2|app-class:VO|business:YES
Traffic Class      : voip-telephony
Business Relevance : business relevant
IP                 : 192.51.100.11

<output truncated>

Switch# show flow exporter option application engines
Engine: prot (IANA_L3_STANDARD, ID: 1)
Engine: port (IANA_L4_STANDARD, ID: 3)
Engine: NBAR (NBAR_CUSTOM, ID: 6)
Engine: cisco (CISCO_L7_GLOBAL, ID: 13)

Switch# show flow exporter option application table
Engine: prot (IANA_L3_STANDARD, ID: 1)
appID  Name      Description

```



```

-----
Engine: port (IANA_L4_STANDARD, ID: 3)
appID  Name      Description
-----
-----

Engine: NBAR (NBAR_CUSTOM, ID: 6)
appID  Name      Description
-----
6:28202 appexample1 User defined protocol appexample1

Engine: cisco (CISCO_L7_GLOBAL, ID: 13)
appID  Name      Description
-----
13:0   unclassified Unclassified traffic
13:1   unknown     Unknown application
13:518 appexample2 appexample2, social web application and service

```

Monitoring AVC with DNS-AS

To display the various AVC with DNS-AS settings you have configured, use these commands in the privileged EXEC mode:

Table 162: AVC with DNS-AS Monitoring Commands

Command	Purpose	Sample Output
show avc dns-as client status	Displays current status of the DNS-AS client. Use this command to know whether AVC with DNS-AS is enabled or not.	Example: show avc dns-as client status
show avc dns-as client trusted-domains	Displays list of trusted domains maintained in the binding table.	Example: show avc dns-as client trusted-domains
show avc dns-as client binding-table and show avc dns-as client binding-table detail	Displays AVC with DNS-AS metadata for the list of trusted domains and resolved entries. You can filter the output by application name, domain name, and so on. Both commands display the same information, in different formats.	Example: show avc dns-as client binding-table
show avc dns-as client statistics	Displays packet logging information—the number of DNS queries sent and the number of responses received.	Example: show avc dns-as client statistics
show avc dns-as client name-server brief	Displays information about the DNS server to which the metadata request was sent.	Example: show avc dns-as client name-server brief

Command	Purpose	Sample Output
show ip name-server	Displays all the name server IP addresses that have been maintained.	Example: show ip name-server
show platform tcam utilization	Displays information about TCAM availability	Example: show platform tcam utilization

Example: show avc dns-as client status

```
Switch# show avc dns-as client status
DNS-AS client is enabled
```

Back to [Table 162: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client trusted-domains

```
Switch# show avc dns-as client trusted-domains
Id | Trusted domain
-----
 1| example.com
 2| www.example.com
 3| example.net
 4| www.example.net
 5| example.org
 6| www.example.org
```

Back to [Table 162: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client binding-table

```
Switch# show avc dns-as client binding-table
Switch# show avc dns-as client binding-table detailed
DNS-AS generated protocols:
Max number of protocols :50
Customization interval [min] :N/A

Age : The amount of time that the entry is active
TTL : Time to live which was learned from DNS-AS server
Time To Expire : Entry expiration time in case device does not see DNS traffic for the entry
host

Protocol-Name : example
VRF : <default>
Host : www.example.com
Age[min] : 2
TTL[min] : 60
Time To Expire[min] : 58
TXT Record : app-name:example|app-class:VO|business:YES
Traffic Class : voip-telephony
Business Relevance : business relevant
IP : 192.0.2.121
: 192.0.2.254
: 198.51.100.1
: 198.51.100.254
: 192.51.100.12
: 203.0.113.125
<output truncated>
```

Back to [Table 162: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client statistics



Note Two DNS servers are configured in this example.

```
Switch# show avc dns-as client statistics
Server details: vrf-id = 0 vrf-name = <default> ip = 192.0.2.1
AAAA Query Error packets 0
AAAA Query TX packets 0
AAAA Response RX packets 0
TXT Query Error packets 0
TXT Query TX packets 8
TXT Response RX packets 0
A Query Error packets 0
A Query TX packets 6
A Response RX packets 0
Server details: vrf-id = 0 vrf-name = <default> ip = 192.0.2.2
AAAA Query Error packets 0
AAAA Query TX packets 0
AAAA Response RX packets 0
TXT Query Error packets 0
TXT Query TX packets 2
TXT Response RX packets 2
A Query Error packets 0
A Query TX packets 4
A Response RX packets 2
Total Drop packets 0

avc_dns_as_pkts_logged = 2
avc_dns_as_q_pkts_processed = 2
```

Back to [Table 162: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client name-server brief

```
Switch# show avc dns-as client name-server brief

Server-IP | Vrf-name
-----
192.0.2.1 | <default>
192.0.2.2 | <default>
```

Back to [Table 162: AVC with DNS-AS Monitoring Commands](#)

Example: show ip name-server

```
Switch# show ip name-server
192.0.2.1
192.0.2.2
2001:DB8::1
```

Back to [Table 162: AVC with DNS-AS Monitoring Commands](#)

Example: show platform team utilization



Note The relevant TCAM entry is IPv4 qos aces:

```
Switch# show platform tcam utilization
CAM Utilization for ASIC# 0 Max Used
Masks/Values Masks/values

Unicast mac addresses: 16604/16604 24/24
IPv4 IGMP groups + multicast routes: 1072/1072 3/3
IPv4 unicast directly-connected routes: 4096/4096 4/4
IPv4 unicast indirectly-connected routes: 1280/1280 40/40
IPv6 Multicast groups: 1072/1072 18/18
IPv6 unicast directly-connected routes: 4096/4096 1/1
IPv6 unicast indirectly-connected routes: 1280/1280 32/32
IPv4 policy based routing aces: 512/512 14/14
IPv4 qos aces: 512/512 51/51
IPv4 security aces: 1024/1024 78/78
IPv6 policy based routing aces: 256/256 8/8
IPv6 qos aces: 256/256 44/44
IPv6 security aces: 512/512 18/18

Note: Allocation of TCAM entries per feature uses
a complex algorithm. The above information is meant
to provide an abstract view of the current TCAM utilization
```

Back to [Table 162: AVC with DNS-AS Monitoring Commands](#)

Troubleshooting AVC with DNS-AS

Problem	Possible Causes and Solutions
There are no entries in the binding table.	The binding table may be empty because of either one or both of these reasons: <ul style="list-style-type: none"> • Metadata is not maintained in DNS server—complete task Generating Metadata Streams, on page 1644 • The entry is not maintained in the trusted domain list—complete task Maintaining the List of Trusted Domains, on page 1647
Unsuccessful DNS snooping or packet logging.	To ensure DNS snooping and packet logging, you must attach the policy map (containing the relevant class maps that will determine traffic class) to the interface—See the example in the Configuring QoS for AVC with DNS-AS, on page 1648
The DNS server does not return correct values.	Verify that the correct DNS-AS metadata is maintained in the DNS system. <ul style="list-style-type: none"> • Using Linux dig: <pre>dig TXT +short www.example.org [dns-server-ip] "CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202"</pre> • Using Windows nslookup: <pre>C:\Windows\system32>NSLookup.exe -q=TXT www.example.org [dns-server-ip] www.example.org text = "CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202"</pre>

Problem	Possible Causes and Solutions
The QoS policy you applied is removed from the port.	<p>When the DNS-AS client recognises an application, along with saving the "A" record response in the binding table, the system utilises the TCAM to save the IP address of the application. A single application can in effect have multiple IP addresses, each utilising additional space in the TCAM. When the TCAM is exhausted, QoS policies cease to be applied.</p> <p>To avoid the problem, monitor TCAM utilisation on a regular basis. Enter the show platform tcam utilisation command in privilege EXEC mode, to display information about TCAM availability.</p>
The DNS-AS client ignores the QoS mapping you've defined and applies default forwarding behavior.	<p>The DNS-AS client ignores QoS mapping and applies default forwarding behavior in these cases:</p> <ul style="list-style-type: none"> • If the match attributes that you specify for the traffic class and business relevance do not match what you have defined in the metadata stream—Check and correct as required. • If the binding table entry is no longer active. This refers to the age of the entry—Use the show avc dns-as client binding-table command to display the age of an entry.

Feature History and Information for AVC with DNS-AS

The following table provides release information about the feature or features described in this chapter. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Release	Modification
Cisco IOS Release 15.2(5)E1	<p>This feature was introduced.</p> <p>Starting with this release, the feature is supported only on the Cisco Catalyst 3560-CX Series Switches and not on the Cisco Catalyst 2960-CX Series Switches.</p>
Cisco IOS Release 15.2(5)E2	<p>Flexible NetFlow (FNF) for AVC with DNS-AS was introduced - Provides the ability to export application information using FNF.</p>



CHAPTER 81

Configuring SDM Templates

- [Finding Feature Information, on page 1663](#)
- [Information About Configuring SDM Templates, on page 1663](#)
- [How to Configure SDM Templates, on page 1665](#)
- [Configuration Examples for SDM Templates, on page 1667](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Information About Configuring SDM Templates

Restrictions for SDM Templates

The following are restrictions when using SDM templates:

SDM Templates

You can use Switch Database Management (SDM) templates to configure system resources to optimize support for specific features, depending on how your device is used in the network.

To allocate ternary content addressable memory (TCAM) resources for different usages, the switch SDM templates prioritize system resources to optimize support for certain features. The templates supported on your device:

- **Default**—The default template gives balance to all functions.



- Note**
- The SDM templates contain only those commands that are defined as part of the templates. If a template enables another related command that is not defined in the template, then this other command will be visible when the **show running config** command is entered. For example, if the SDM template enables the **switchport voice vlan** command, then the **spanning-tree portfast edge** command may also be enabled (although it is not defined on the SDM template).

If the SDM template is removed, then other such related commands are also removed and have to be reconfigured explicitly.
 - SDM templates do not create VLANs. You must create the VLANs before adding commands to the SDM templates.

Default Templates for Catalyst 2960-CX

The templates for Catalyst 2960-CX switches are applicable for the LAN Base license.

Table 163: Approximate Number of Feature Resources Allowed by Templates

Resource	Default
Unicast MAC addresses	16K
Active VLANs/VLAN IDs	255/4096
NetFlow entries	16K
Etherchannel groups per stack	6
IPv4 IGMP or IPv6 groups	1K IPv4 1K IPv6
Direct routes	2K IPv4 2K IPv6
Indirect routes	1K IPv4 1K IPv6 (16 static routes only)
IPv4 or IPv6 policy-based routing ACEs	0 (IPv4 PBR) 0 (IPv6 PBR)
IPv4 or IPv6 MAC QoS ACEs	0.375K (IPv4 QoS) 0.25K (IPv6 QoS)
IPv4 or IPv6 port or MAC security ACEs	0.375K (IPv4 ACL) 0.375K (IPv6 ACL)

Default Templates for Catalyst 3560-CX

The templates for Catalyst 3560-CX switches are applicable for IP Base and IP Services licenses.

Table 164: Approximate Number of Feature Resources Allowed by Templates

Resource	Default
Unicast MAC addresses	16K
Active VLANs/VLAN IDs	1K/4096
Etherchannel groups per stack	6
IPv4 IGMP or IPv6 groups	1K IPv4 1K IPv6
Direct routes	4K IPv4 4K IPv6
Indirect routes	1K IPv4 1K IPv6
IPv4 or IPv6 policy-based routing ACEs	0.25K (IPv4 PBR) 0.25K (IPv6 PBR)
IPv4 or IPv6 QoS ACEs	0.375K (IPv4 QoS) 0.25K (IPv6 QoS)
IPv4 or IPv6 port or MAC security ACEs	0.375K (IPv4 ACL) 0.375K (IPv6 ACL)

How to Configure SDM Templates

Setting the SDM Template

Follow these steps to use the SDM template to maximize feature usage:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `sdm prefer { advanced | vlan }`
4. `sdm prefer { default }`
5. `end`
6. `reload`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	sdm prefer { advanced vlan } Example: Switch(config)# sdm prefer advanced	Specifies the SDM template to be used on the switch. The keywords have these meanings: <ul style="list-style-type: none"> • advanced —Supports advanced features such as Netflow. • vlan —Maximizes VLAN configuration on the switch with no routing supported in hardware. <p>Note The no sdm prefer command and a default template is not supported.</p>
Step 4	sdm prefer { default } Example: Switch(config)# sdm prefer lanbase-routing	Specifies the SDM template to be used on the switch. The keywords have these meanings: <ul style="list-style-type: none"> • default—The default template provides balance for all Layer 2, IPv4 and IPv6 functionality. <p>Use the no sdm prefer command to set the switch to the default template, The default template balances the use of system resources.</p>
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	reload Example: Switch# reload	Reloads the operating system.

Configuration Examples for SDM Templates

Examples: Displaying SDM Templates

This is an example output showing the default template information.

Example output showing the default template information on a Catalyst 3560-CX switch.

```
Device# show sdm prefer
```

```
The current template is "default" template.
The selected template optimizes the resources in
the switch to support this level of features for
8 routed interfaces and 1024 VLANs.
```

```

number of unicast mac addresses:          16K
number of IPv4 IGMP groups + multicast routes: 1K
number of IPv4 unicast routes:           5K
number of directly-connected IPv4 hosts:  4K
number of indirect IPv4 routes:          1K
number of IPv6 multicast groups:         1K
number of IPv6 unicast routes:           5K
number of directly-connected IPv6 addresses: 4K
number of indirect IPv6 unicast routes:   1K
number of IPv4 policy based routing aces: 0.25K
number of IPv4/MAC qos aces:             0.375k
number of IPv4/MAC security aces:        0.375k
number of IPv6 policy based routing aces: 0.25K
number of IPv6 qos aces:                 0.25K
number of IPv6 security aces:            0.375k

```

Example output showing the default template information on a Catalyst 2960-CX switch.

```
Device# show sdm prefer
```

```
The current template is "default" template.
The selected template optimizes the resources in
the switch to support this level of features for
0 routed interfaces and 255 VLANs.
```

```

number of unicast mac addresses:          16K
number of IPv4 IGMP groups + multicast routes: 1K
number of IPv4 unicast routes:           3K
number of directly-connected IPv4 hosts:  2K
number of indirect IPv4 routes:          1K
number of IPv6 multicast groups:         1K
number of IPv6 unicast routes:           3K
number of directly-connected IPv6 addresses: 2K
number of indirect IPv6 unicast routes:   1K
number of IPv4 policy based routing aces: 0
number of IPv4/MAC qos aces:             0.375k
number of IPv4/MAC security aces:        0.375k
number of IPv6 policy based routing aces: 0
number of IPv6 qos aces:                 0.25K
number of IPv6 security aces:            0.375k

```

Examples: Configuring SDM Templates

This example shows how to configure the VLAN template:

```
Switch(config)# sdm prefer lanbase-routing  
Switch(config)# exit  
Switch# reload  
  Proceed with reload? [confirm]
```



CHAPTER 82

Configuring System Message Logs

- [Restrictions for Configuring System Message Logs, on page 1669](#)
- [Information About Configuring System Message Logs, on page 1669](#)
- [How to Configure System Message Logs, on page 1672](#)
- [Monitoring and Maintaining System Message Logs, on page 1680](#)
- [Configuration Examples for System Message Logs, on page 1680](#)

Restrictions for Configuring System Message Logs

When the **logging discriminator** command is configured, the device may experience memory leak or crash. This usually happens during heavy syslog or debug output. The rate of the memory leak is dependent on the number of logs being produced. In extreme cases, the device may also crash. As a workaround, use the **no logging discriminator** command to disable the logging discriminator.

Information About Configuring System Message Logs

System Message Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. . The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration. The process also sends messages to the console.

When the logging process is disabled, messages are sent only to the console. The messages are sent as they are generated, so message and debug output are interspersed with prompts or output from other commands. Messages appear on the active consoles after the process that generated them has finished.

You can set the severity level of the messages to control the type of messages displayed on the consoles and each of the destinations. You can time-stamp log messages or set the syslog source address to enhance real-time debugging and management. For information on possible messages, see the system message guide for this release.

You can access logged system messages by using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves syslog messages in an internal buffer on a standalone switch. If a standalone switch, the log is lost unless you had saved it to flash memory.

You can remotely monitor system messages by viewing the logs on a syslog server or by accessing the switch through Telnet, through the console port, or through the Ethernet management port.



Note The syslog format is compatible with 4.3 BSD UNIX.

System Log Message Format

System log messages can contain up to 80 characters and a percent sign (%), which follows the optional sequence number or time-stamp information, if configured. Depending on the switch, messages appear in one of these formats:

- *seq no:timestamp: %facility-severity-MNEMONIC:description (hostname-n)*
- *seq no:timestamp: %facility-severity-MNEMONIC:description*

The part of the message preceding the percent sign depends on the setting of these global configuration commands:

- **service sequence-numbers**
- **service timestamps log datetime**
- **service timestamps log datetime [localtime] [msec] [show-timezone]**
- **service timestamps log uptime**

Table 165: System Log Message Elements

Element	Description
<i>seq no:</i>	Stamps log messages with a sequence number only if the service sequence-numbers global configuration command is configured.
<i>timestamp</i> formats: <i>mm/dd h h:mm:ss</i> or <i>hh:mm:ss</i> (short uptime) or <i>d h</i> (long uptime)	Date and time of the message or event. This information appears only if the service timestamps log [datetime log] global configuration command is configured.
<i>facility</i>	The facility to which the message refers (for example, SNMP, SYS, and so forth).
<i>severity</i>	Single-digit code from 0 to 7 that is the severity of the message.
<i>MNEMONIC</i>	Text string that uniquely describes the message.
<i>description</i>	Text string containing detailed information about the event being reported.

Default System Message Logging Settings

Table 166: Default System Message Logging Settings

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging.
Logging file configuration	No filename specified.
Logging buffer size	4096 bytes.
Logging history size	1 message.
Time stamps	Disabled.
Synchronous logging	Disabled.
Logging server	Disabled.
Syslog server IP address	None configured.
Server facility	Local7
Server severity	Informational.

Enabling Syslog Trap Messages

You can enable Syslog traps using the **snmp-server enable traps syslog** command.

After enabling Syslog traps, you have to specify the trap message severity. Use the **logging snmp-trap** command to specify the trap level. By default, the command enables severity 0 to 4. To enable all the severity level, configure the **logging snmp-trap 0 7** command.

To enable individual trap levels, configure the following commands:

- **logging snmp-trap emergencies**: Enables only severity 0 traps.
- **logging snmp-trap alert** Enables only severity 1 traps.

Note that, along with the Syslog traps, the Syslog history should also be applied. Without this configuration, Syslog traps are not sent.

Use the **logging history informational** command to enable the Syslog history.

How to Configure System Message Logs

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging buffered** [*size*]
3. **logging** *host*
4. **logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*]
5. **end**
6. **terminal monitor**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	logging buffered [<i>size</i>] Example: Switch(config)# logging buffered 8192	<p>Logs messages to an internal buffer on the switch or on a standalone switch or, in the case of a switch stack, on the active switch. The range is 4096 to 2147483647 bytes. The default buffer size is 4096 bytes.</p> <p>If a standalone switch or the active switch fails, the log file is lost unless you previously saved it to flash memory. See Step 4.</p> <p>Note Do not make the buffer size too large because the switch could run out of memory for other tasks. Use the show memory privileged EXEC command to view the free processor memory on the switch. However, this value is the maximum available, and the buffer size should <i>not</i> be set to this amount.</p>
Step 3	logging <i>host</i> Example: Switch(config)# logging 125.1.1.100	<p>Logs messages to a UNIX syslog server host.</p> <p><i>host</i> specifies the name or IP address of the host to be used as the syslog server.</p> <p>To build a list of syslog servers that receive logging messages, enter this command more than once.</p>

	Command or Action	Purpose
Step 4	<p>logging file flash: <i>filename</i> [<i>max-file-size</i> [<i>min-file-size</i>]] [<i>severity-level-number</i> <i>type</i>]</p> <p>Example:</p> <pre>Switch(config)# logging file flash:log_msg.txt 40960 4096 3</pre>	<p>Stores log messages in a file in flash memory on a standalone switch or, in the case of a switch stack, on the active switch.</p> <ul style="list-style-type: none"> • <i>filename</i>—Enters the log message filename. • (Optional) max-file-size —Specifies the maximum logging file size. The range is 4096 to 2147483647. The default is 4096 bytes. • (Optional) <i>min-file-size</i>—Specifies the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) <i>severity-level-number</i> <i>type</i>—Specifies either the logging severity level or the logging type. The severity range is 0 to 7.
Step 5	<p>end</p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p>terminal monitor</p> <p>Example:</p> <pre>Switch# terminal monitor</pre>	<p>Logs messages to a nonconsole terminal during the current session.</p> <p>Terminal parameter-setting commands are set locally and do not remain in effect after the session has ended. You must perform this step for each session to see the debugging messages.</p>

Synchronizing Log Messages

You can synchronize unsolicited messages and **debug** privileged EXEC command output with solicited device output and prompts for a specific console port line or virtual terminal line. You can identify the types of messages to be output asynchronously based on the level of severity. You can also configure the maximum number of buffers for storing asynchronous messages for the terminal after which messages are dropped.

When synchronous logging of unsolicited messages and **debug** command output is enabled, unsolicited device output appears on the console or printed after solicited device output appears or is printed. Unsolicited messages and **debug** command output appears on the console after the prompt for user input is returned. Therefore, unsolicited messages and **debug** command output are not interspersed with solicited device output and prompts. After the unsolicited messages appear, the console again displays the user prompt.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **line** [**console** | **vty**] *line-number* [*ending-line-number*]

3. **logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>line [console vty] <i>line-number</i> [<i>ending-line-number</i>]</p> <p>Example:</p> <pre>Switch(config)# line console</pre>	<p>Specifies the line to be configured for synchronous logging of messages.</p> <ul style="list-style-type: none"> • console—Specifies configurations that occur through the switch console port or the Ethernet management port. • line vty <i>line-number</i>—Specifies which vty lines are to have synchronous logging enabled. You use a vty connection for configurations that occur through a Telnet session. The range of line numbers is from 0 to 15. <p>You can change the setting of all 16 vty lines at once by entering:</p> <pre>line vty 0 15</pre> <p>You can also change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter:</p> <pre>line vty 2</pre> <p>When you enter this command, the mode changes to line configuration.</p>
Step 3	<p>logging synchronous [level [<i>severity-level</i> all] limit <i>number-of-buffers</i>]</p> <p>Example:</p> <pre>Switch(config)# logging synchronous level 3 limit 1000</pre>	<p>Enables synchronous logging of messages.</p> <ul style="list-style-type: none"> • (Optional) level <i>severity-level</i>—Specifies the message severity level. Messages with a severity level equal to or higher than this value are printed asynchronously. Low numbers mean greater severity and high numbers mean lesser severity. The default is 2. • (Optional) level all—Specifies that all messages are printed asynchronously regardless of the severity level. • (Optional) limit <i>number-of-buffers</i>—Specifies the number of buffers to be queued for the terminal after which new messages are dropped. The range is 0 to 2147483647. The default is 20.

	Command or Action	Purpose
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Disabling Message Logging

Message logging is enabled by default. It must be enabled to send messages to any destination other than the console. When enabled, log messages are sent to a logging process, which logs messages to designated locations asynchronously to the processes that generated the messages.

Disabling the logging process can slow down the switch because a process must wait until the messages are written to the console before continuing. When the logging process is disabled, messages appear on the console as soon as they are produced, often appearing in the middle of command output.

The **logging synchronous** global configuration command also affects the display of messages to the console. When this command is enabled, messages appear only after you press **Return**.

To reenable message logging after it has been disabled, use the **logging on** global configuration command.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **no logging console**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	no logging console Example: <pre>Switch(config)# no logging console</pre>	Disables message logging.
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Enabling and Disabling Time Stamps on Log Messages

By default, log messages are not time-stamped.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of these commands:
 - **service timestamps log uptime**
 - **service timestamps log datetime[msec | localtime | show-timezone]**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	Use one of these commands: <ul style="list-style-type: none"> • service timestamps log uptime • service timestamps log datetime[msec localtime show-timezone] Example: Switch(config)# service timestamps log uptime or Switch(config)# service timestamps log datetime	Enables log time stamps. <ul style="list-style-type: none"> • log uptime—Enables time stamps on log messages, showing the time since the system was rebooted. • log datetime—Enables time stamps on log messages. Depending on the options selected, the time stamp can include the date, time in milliseconds relative to the local time zone, and the time zone name.
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Enabling and Disabling Sequence Numbers in Log Messages

If there is more than one log message with the same time stamp, you can display messages with sequence numbers to view these messages. By default, sequence numbers in log messages are not displayed.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **service sequence-numbers**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	service sequence-numbers Example: <pre>Switch(config)# service sequence-numbers</pre>	Enables sequence numbers.
Step 3	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Defining the Message Severity Level

Limit messages displayed to the selected device by specifying the severity level of the message.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging console *level***
3. **logging monitor *level***
4. **logging trap *level***
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	logging console <i>level</i> Example: Switch(config) # logging console 3	Limits messages logged to the console. By default, the console receives debugging messages and numerically lower levels.
Step 3	logging monitor <i>level</i> Example: Switch(config) # logging monitor 3	Limits messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels.
Step 4	logging trap <i>level</i> Example: Switch(config) # logging trap 3	Limits messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels.
Step 5	end Example: Switch(config) # end	Returns to privileged EXEC mode.

Limiting Syslog Messages Sent to the History Table and to SNMP

This task explains how to limit syslog messages that are sent to the history table and to SNMP.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging history** *level*
3. **logging history size** *number*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	logging history level Example: <pre>Switch(config)# logging history 3</pre>	Changes the default level of syslog messages stored in the history file and sent to the SNMP server. By default, warnings , errors , critical , alerts , and emergencies messages are sent.
Step 3	logging history size number Example: <pre>Switch(config)# logging history size 200</pre>	Specifies the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 0 to 500 messages.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.

Logging Messages to a UNIX Syslog Daemon

This task is optional.



Note Some recent versions of UNIX syslog daemons no longer accept by default syslog packets from the network. If this is the case with your system, use the UNIX **man syslogd** command to decide what options must be added to or removed from the syslog command line to enable logging of remote syslog messages.

Before you begin

- Log in as root.
- Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server.

SUMMARY STEPS

1. Add a line to the file `/etc/syslog.conf`.
2. Enter these commands at the UNIX shell prompt.
3. Make sure the syslog daemon reads the new changes.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Add a line to the file <code>/etc/syslog.conf</code> . Example:	<ul style="list-style-type: none"> • local7—Specifies the logging facility.

	Command or Action	Purpose
	<code>local7.debug /usr/adm/logs/cisco.log</code>	<ul style="list-style-type: none"> debug—Specifies the syslog level. The file must already exist, and the syslog daemon must have permission to write to it.
Step 2	Enter these commands at the UNIX shell prompt. Example: <pre>\$ touch /var/log/cisco.log \$ chmod 666 /var/log/cisco.log</pre>	Creates the log file. The syslog daemon sends messages at this level or at a more severe level to this file.
Step 3	Make sure the syslog daemon reads the new changes. Example: <pre>\$ kill -HUP `cat /etc/syslog.pid`</pre>	For more information, see the man syslog.conf and man syslogd commands on your UNIX system.

Monitoring and Maintaining System Message Logs

Monitoring Configuration Archive Logs

Command	Purpose
<pre>show archive log config {all number [<i>end-number</i>] user <i>username</i> [<i>session number</i>] number [<i>end-number</i>] statistics} [provisioning]</pre>	Displays the entire configuration log or the log for specified parameters.

Configuration Examples for System Message Logs

Example: Switch System Message

This example shows a partial switch system message on a switch:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channell1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state
to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```




CHAPTER 83

Configuring Online Diagnostics

- [Information About Configuring Online Diagnostics, on page 1681](#)
- [How to Configure Online Diagnostics, on page 1682](#)
- [Monitoring and Maintaining Online Diagnostics, on page 1686](#)
- [Configuration Examples for Online Diagnostic Tests, on page 1687](#)

Information About Configuring Online Diagnostics

Online Diagnostics

With online diagnostics, you can test and verify the hardware functionality of the Device while the Device is connected to a live network.

The online diagnostics contain packet switching tests that check different hardware components and verify the data path and the control signals.

The online diagnostics detect problems in these areas:

- Hardware components
- Interfaces (Ethernet ports and so forth)
- Solder joints

Online diagnostics are categorized as on-demand, scheduled, or health-monitoring diagnostics. On-demand diagnostics run from the CLI; scheduled diagnostics run at user-designated intervals or at specified times when the Device is connected to a live network; and health-monitoring runs in the background with user-defined intervals. By default, the health-monitoring test runs for every 30 seconds.

After you configure online diagnostics, you can manually start diagnostic tests or display the test results. You can also see which tests are configured for the Device or switch stack and the diagnostic tests that have already run.

How to Configure Online Diagnostics

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the switch, use the **diagnostic start** privileged EXEC command to begin diagnostic testing.

After starting the tests, you cannot stop the testing process.

Use this privileged EXEC command to manually start online diagnostic testing.

SUMMARY STEPS

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** }

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>diagnostic start switch <i>number</i> test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic non-disruptive }</p> <p>Example:</p> <pre>Switch# diagnostic start switch 2 test basic</pre>	<p>Starts the diagnostic tests.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>You can specify the tests by using one of these options:</p> <ul style="list-style-type: none"> • <i>name</i>—Enters the name of the test. • <i>test-id</i>—Enters the ID number of the test. • <i>test-id-range</i>—Enters the range of test IDs by using integers separated by a comma and a hyphen. • all—Starts all of the tests. • basic— Starts the basic test suite. • non-disruptive—Starts the non-disruptive test suite.

Configuring Online Diagnostics

You must configure the failure threshold and the interval between tests before enabling diagnostic monitoring.

Scheduling Online Diagnostics

You can schedule online diagnostics to run at a designated time of day or on a daily, weekly, or monthly basis for a switch. Use the **no** form of this command to remove the scheduling.

SUMMARY STEPS

1. **configure terminal**

2. **diagnostic schedule switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** } {**daily** | **on** *mm dd yyyy hh:mm* | **weekly** *day-of-week hh:mm*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>diagnostic schedule switch <i>number</i> test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic non-disruptive } {daily on <i>mm dd yyyy hh:mm</i> weekly <i>day-of-week hh:mm</i>}</p> <p>Example:</p> <pre>Switch(config)# diagnostic schedule switch 1 test 1-5 on July 3 2013 23:10</pre>	<p>Schedules on-demand diagnostic tests for a specific day and time.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>When specifying the tests to be scheduled, use these options:</p> <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All test IDs. • basic—Starts the basic on-demand diagnostic tests. • non-disruptive—Starts the non-disruptive test suite. <p>You can schedule the tests as follows:</p> <ul style="list-style-type: none"> • Daily—Use the daily <i>hh:mm</i> parameter. • Specific day and time—Use the on <i>mm dd yyyy hh:mm</i> parameter. • Weekly—Use the weekly <i>day-of-week hh:mm</i> parameter.

Configuring Health-Monitoring Diagnostics

You can configure health-monitoring diagnostic testing on a Device while it is connected to a live network. You can configure the execution interval for each health-monitoring test, enable the Device to generate a syslog message because of a test failure, and enable a specific test.

Use the **no** form of this command to disable testing.

By default, health monitoring is disabled, but the Device generates a syslog message when a test fails.

Follow these steps to configure and enable the health-monitoring diagnostic tests:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **diagnostic monitor interval switch** *number test* {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*
4. **diagnostic monitor syslog**
5. **diagnostic monitor threshold switch** *number number test* {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*
6. **diagnostic monitor switch** *number test* {*name* | *test-id* | *test-id-range* | **all**}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	diagnostic monitor interval switch <i>number test</i> { <i>name</i> <i>test-id</i> <i>test-id-range</i> all } <i>hh:mm:ss milliseconds day</i> Example: <pre>Switch(config)# diagnostic monitor interval switch 2 test 1 12:30:00 750 5</pre>	Configures the health-monitoring interval of the specified tests. <p>When specifying the tests, use one of these parameters:</p> <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests. <p>When specifying the interval, set these parameters:</p> <ul style="list-style-type: none"> • <i>hh:mm:ss</i>—Monitoring interval in hours, minutes, and seconds. The range for <i>hh</i> is 0 to 24, and the range for <i>mm</i> and <i>ss</i> is 0 to 60.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>milliseconds</i>—Monitoring interval in milliseconds (ms). The range is from 0 to 999. • <i>day</i>—Monitoring interval in the number of days. The range is from 0 to 20.
Step 4	<p>diagnostic monitor syslog</p> <p>Example:</p> <pre>Switch(config)# diagnostic monitor syslog</pre>	(Optional) Configures the switch to generate a syslog message when a health-monitoring test fails.
Step 5	<p>diagnostic monitor threshold switch <i>number number test</i> {<i>name</i> <i>test-id</i> <i>test-id-range</i> all} failure count count</p> <p>Example:</p> <pre>Switch(config)# diagnostic monitor threshold switch 2 test 1 failure count 20</pre>	<p>(Optional) Sets the failure threshold for the health-monitoring tests.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>When specifying the tests, use one of these parameters:</p> <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests. <p>The range for the failure threshold <i>count</i> is 0 to 99.</p>
Step 6	<p>diagnostic monitor switch <i>number test</i> {<i>name</i> <i>test-id</i> <i>test-id-range</i> all}</p> <p>Example:</p> <pre>Switch(config)# diagnostic monitor switch 2 test 1</pre>	<p>Enables the specified health-monitoring tests.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>When specifying the tests, use one of these parameters:</p> <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests.
Step 7	<p>end</p> <p>Example:</p>	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	
Step 8	show running-config Example: Switch# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no diagnostic monitor interval test***test-id | test-id-range* } global configuration command to change the interval to the default value or to zero. Use the **no diagnostic monitor syslog** command to disable generation of syslog messages when a health-monitoring test fails. Use the **diagnostic monitor threshold test***test-id | test-id-range* }**failure count** command to remove the failure threshold.

Monitoring and Maintaining Online Diagnostics

Displaying Online Diagnostic Tests and Test Results

You can display the online diagnostic tests that are configured for the Device or Device stack and check the test results by using the privileged EXEC **show** commands in this table:

Table 167: Commands for Diagnostic Test Configuration and Results

Command	Purpose
show diagnostic content switch [<i>number</i> all]	Displays the online diagnostics configured for a switch.
show diagnostic status	Displays the currently running diagnostic tests.
show diagnostic result switch [<i>number</i> all] [detail test { <i>name</i> <i>test-id</i> <i>test-id-range</i> all } [detail]]	Displays the online diagnostics test results.
show diagnostic switch [<i>number</i> all] [detail]	Displays the online diagnostics test results.
show diagnostic schedule switch [<i>number</i> all]	Displays the online diagnostics test schedule.
show diagnostic post	Displays the POST results. (The output is the same as the show post command output.)

Configuration Examples for Online Diagnostic Tests

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the switch, use the **diagnostic start** privileged EXEC command to begin diagnostic testing.

After starting the tests, you cannot stop the testing process.

Use this privileged EXEC command to manually start online diagnostic testing.

SUMMARY STEPS

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** }

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>diagnostic start switch <i>number</i> test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic non-disruptive }</p> <p>Example:</p> <pre>Switch# diagnostic start switch 2 test basic</pre>	<p>Starts the diagnostic tests.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>You can specify the tests by using one of these options:</p> <ul style="list-style-type: none"> • <i>name</i>—Enters the name of the test. • <i>test-id</i>—Enters the ID number of the test. • <i>test-id-range</i>—Enters the range of test IDs by using integers separated by a comma and a hyphen. • all—Starts all of the tests. • basic— Starts the basic test suite. • non-disruptive—Starts the non-disruptive test suite.

Example: Configure a Health Monitoring Test

This example shows how to configure a health-monitoring test:

```
Switch(config)# diagnostic monitor threshold switch 1 test 1 failure count 50
Switch(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

Examples: Schedule Diagnostic Test

This example shows how to schedule diagnostic testing for a specific day and time on a specific switch:

```
Switch(config)# diagnostic schedule test DiagThermalTest on June 3 2013 22:25
```

This example shows how to schedule diagnostic testing to occur weekly at a certain time on a specific switch:

```
Switch(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

Displaying Online Diagnostics: Examples

This example shows how to display the online diagnostic detailed information on a specific switch:

```
Switch# show diagnostic switch 1 detail
```

```
Switch 1: SerialNo :
```

```
Overall Diagnostic Result for Switch 1 : UNTESTED
```

```
Test results: (. = Pass, F = Fail, U = Untested)
```

```
1) TestPortAsicStackPortLoopback ---> U
```

```
Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0
```

```
2) TestPortAsicLoopback -----> U
```

```
Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0
```

```
3) TestPortAsicCam -----> U
```

```
Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0
```

```

4) TestPortAsicMem -----> U

Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0

```

```

5) TestInlinePwrCtrlr -----> U

Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0

```

This example shows how to display the online diagnostics that are configured on a specific switch:

```
Switch# show diagnostic content switch 3
```

```
Switch 1:
```

```
Diagnostics test suite attributes:
```

```

B/* - Basic ondemand test / NA
P/V/* - Per port test / Per device test / NA
D/N/* - Disruptive test / Non-disruptive test / NA
S/* - Only applicable to standby unit / NA
X/* - Not a health monitoring test / NA
F/* - Fixed monitoring interval test / NA
E/* - Always enabled monitoring test / NA
A/I - Monitoring is active / Monitoring is inactive
R/* - Switch will reload after test list completion / NA
P/* - will partition stack / NA

```

ID	Test Name	Attributes	Test Interval day hh:mm:ss.ms	Three- day shold
1)	TestPortAsicStackPortLoopback	B*N***I**	not configured	n/a
2)	TestPortAsicLoopback	B*D*X**IR*	not configured	n/a
3)	TestPortAsicCam	B*D*X**IR*	not configured	n/a
4)	TestPortAsicRingLoopback	B*D*X**IR*	not configured	n/a
5)	TestMicRingLoopback	B*D*X**IR*	not configured	n/a
6)	TestPortAsicMem	B*D*X**IR*	not configured	n/a

This example shows how to display the online diagnostic results for a switch:

```
Switch# show diagnostic result
```

```
Switch 1: SerialNo :
Overall diagnostic result: PASS
```

```

Test results: (. = Pass, F = Fail, U = Untested)
1) TestPortAsicStackPortLoopback ---> .
2) TestPortAsicLoopback -----> .
3) TestPortAsicCam -----> .
4) TestPortAsicRingLoopback -----> .
5) TestMicRingLoopback -----> .
6) TestPortAsicMem -----> .

```

This example shows how to display the online diagnostic test status:

```
Switch# show diagnostic status
```

```

<BU> - Bootup Diagnostics, <HM> - Health Monitoring Diagnostics,
<OD> - OnDemand Diagnostics, <SCH> - Scheduled Diagnostics
=====
Card   Description                               Current Running Test           Run by
-----
1      N/A                                         N/A                             N/A
2      TestPortAsicStackPortLoopback             <OD>
      TestPortAsicLoopback                     <OD>
      TestPortAsicCam                           <OD>
      TestPortAsicRingLoopback                 <OD>
      TestMicRingLoopback                      <OD>
      TestPortAsicMem                           <OD>
3      N/A                                         N/A                             N/A
4      N/A                                         N/A                             N/A
=====
Switch#

```

This example shows how to display the online diagnostic test schedule for a switch:

```
Switch# show diagnostic schedule switch 1
```

```

Current Time = 14:39:49 PST Tue May 5 2013
Diagnostic for Switch 1:
Schedule #1:
To be run daily 12:00
Test ID(s) to be executed: 1.

```



CHAPTER 84

Data Sanitization

This module provides information on how to sanitize data from devices.

- [Data Sanitization, on page 1691](#)

Data Sanitization

Use the National Institute of Standards and Technology (NIST) purge method that renders the data unrecoverable through simple, non-invasive data recovery techniques or through state-of-the-art laboratory techniques.



Note Unless otherwise stated, the data sanitization instructions provide NIST 800-88 clear sanitization techniques in user-addressable storage locations for protection against simple non-invasive data recovery techniques and do not provide techniques that render data recovery infeasible using state of the art laboratory techniques.

Follow these steps to remove the files from a flash drive:

Step 1 **factory-reset all secure**

Example:

```
Device> factory-reset all secure
```

Purges the data on the flash.

Step 2 Copy the image to the flash using TFTP.

For more information, see [Copying Image Files using TFTP](#).

Step 3 **reload**

Example:

```
Device> reload
```

Reloads the device.

Note If you have copied the image to the flash drive (Step 2), the switch reboots automatically.

Step 4 **show platform software factory-reset secure log**

Example:

```
Device> show platform software factory-reset secure log
```

Displays the data sanitization report.

Example: Data Sanitization

The following example shows how to reset all data from a device:

```
Device# factory-reset all secure
```

```
The factory reset operation is irreversible for all operations. Are you sure? [confirm]
```

```
The following will be deleted as a part of factory reset: NIST-SP-800-88-R1
```

- 1: Crash info and logs
- 2: User data, startup and running configuration
- 3: All IOS images, including the current boot image
- 4: User added rommon variables
- 5: OBFL logs
- 6: License usage log files

Note:

1. You are advised to COPY an IOS image via TFTP after factory-reset and before reloading the box (OPTIONAL)
2. Then, Reload the box for factory-reset to complete

```
DO NOT UNPLUG THE POWER OR INTERRUPT THE OPERATION
```

```
Are you sure you want to continue?
```

```
[confirm]
```

```
% factory-reset: started.
% Format of nvram start..
% Format of nvram end...
```

```
*Sep 20 11:36:14.980: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram
```

```
% Erase of obfl0 start...
```

```
.....
```

```
% Erase of obfl0 end...
```

```
% Validating obfl0 partition...
```

```
00000000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

```
.....
```

```
003FFFF0: **
```

```
.
```

```
% Format of obfl0 start
% Format of obfl0 complete
% Erase of rsvd start...
```

```

.....

% Erase of rsvd end...
% Validating rsvd partition...

00000000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

.....

000DFFF0: **

.

% Erase of flash start...

.....

% Erase of flash end...
% Validating flash partition...

00000000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

.....

0E9FFFF0: **

.

% Format of flash start
% Format of flash complete
% Format of vb: start...
% Format of vb: end...
% act2 erase started...

----- USER 1 -----

ObjectID  ObjectType  ObjectSize
=====

0xBA7E1F05  0x01        0x00DC

% act2 erase completed...

#CISCO C1000-48T-4G-L DATA SANITIZATION REPORT#

START : 2022-09-20 11:36:11
END   : 2022-09-20 11:37:28
PNM   : NAND
MNM   : IS34/35ML02G084
MID   : 0x00
DID   : 0xDAC8
NIST  : PURGE SUCCESS

% factory-reset: logging success...
% FACTORY-RESET - Secure Successfull...

1. You are advised to COPY an IOS image via TFTP before reloading the box (OPTIONAL)
2. Then, Reload the box for factory-reset to complete

```

The following is sample output from the show platform software factory-reset secure log command after a secure factory reset of the device:

```
Device# show platform software factory-reset secure log
```

```
#CISCO C1000-48T-4G-L DATA SANITIZATION REPORT#  
START : 2022-07-13 10:50:29  
END   : 2022-07-13 10:51:45  
PNM   : NAND  
MNM   : IS34/35ML02G084  
MID   : 0x00  
DID   : 0xDAC8  
NIST  : PURGE SUCCESS
```



CHAPTER 85

Troubleshooting the Software Configuration

This chapter describes how to identify and resolve software problems related to the Cisco IOS software on the switch. Depending on the nature of the problem, you can use the command-line interface (CLI), Device Manager, or Network Assistant to identify and solve problems.

Additional troubleshooting information, such as LED descriptions, is provided in the hardware installation guide.

- [Information About Troubleshooting the Software Configuration, on page 1695](#)
- [How to Troubleshoot the Software Configuration, on page 1701](#)
- [Verifying Troubleshooting of the Software Configuration, on page 1715](#)
- [Scenarios for Troubleshooting the Software Configuration, on page 1718](#)
- [Configuration Examples for Troubleshooting Software, on page 1720](#)

Information About Troubleshooting the Software Configuration

Software Failure on a Switch

Switch software can be corrupted during an upgrade by downloading the incorrect file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

Lost or Forgotten Password on a Device

The default configuration for the device allows an end user with physical access to the device to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the device.



Note On these devices, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message reminds you to return to the default configuration during the recovery process.



Note You cannot recover encryption password key, when Cisco WLC configuration is copied from one Cisco WLC to another (in case of an RMA).

Power over Ethernet Ports

A Power over Ethernet (PoE) switch port automatically supplies power to one of these connected devices if the switch detects that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device
- an IEEE 802.3at-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

After the switch detects a powered device, the switch determines the device power requirements and then grants or denies power to the device. The switch can also detect the real-time power consumption of the device by monitoring and policing the power usage.

Disabled Port Caused by Power Loss

If a powered device (such as a Cisco IP Phone 7910) that is connected to a PoE Device port and powered by an AC power source loses power from the AC power source, the device might enter an error-disabled state. To recover from an error-disabled state, enter the **shutdown** interface configuration command, and then enter the **no shutdown** interface command. You can also configure automatic recovery on the Device to recover from the error-disabled state.

On a Device, the **errdisable recovery cause loopback** and the **errdisable recovery interval seconds** global configuration commands automatically take the interface out of the error-disabled state after the specified period of time.

Monitoring PoE Port Status

- **show controllers power inline** privileged EXEC command
- **show power inline** EXEC command
- **debug ilpower** privileged EXEC command

Disabled Port Caused by False Link-Up

If a Cisco powered device is connected to a port and you configure the port by using the **power inline never** interface configuration command, a false link-up can occur, placing the port into an error-disabled state. To take the port out of the error-disabled state, enter the **shutdown** and the **no shutdown** interface configuration commands.

You should not connect a Cisco powered device to a port that has been configured with the **power inline never** command.

Ping

The Device supports IP ping, which you can use to test connectivity to remote hosts. Ping sends an echo request packet to an address and waits for a reply. Ping returns one of these responses:

- Normal response—The normal response (*hostname is alive*) occurs in 1 to 10 seconds, depending on network traffic.
- Destination does not respond—If the host does not respond, a *no-answer* message is returned.
- Unknown host—If the host does not exist, an *unknown host* message is returned.
- Destination unreachable—If the default gateway cannot reach the specified network, a *destination-unreachable* message is returned.
- Network or host unreachable—If there is no entry in the route table for the host or network, a *network or host unreachable* message is returned.

Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses. Traceroute finds the path by using the MAC address tables of the Device in the path. When the Device detects a device in the path that does not support Layer 2 traceroute, the Device continues to send Layer 2 trace queries and lets them time out.

The Device can only identify the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

Layer 2 Traceroute Guidelines

- Cisco Discovery Protocol (CDP) must be enabled on all the devices in the network. For Layer 2 traceroute to function properly, do not disable CDP.
If any devices in the physical path are transparent to CDP, the switch cannot identify the path through these devices.
- A Device is reachable from another Device when you can test connectivity by using the **ping** privileged EXEC command. All Device in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command on a Device that is not in the physical path from the source device to the destination device. All Device in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.
- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.

- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.
- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the Device uses the Address Resolution Protocol (ARP) to associate the IP addresses with the corresponding MAC addresses and the VLAN IDs.
 - If an ARP entry exists for the specified IP address, the Device uses the associated MAC address and identifies the physical path.
 - If an ARP entry does not exist, the Device sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.
- This feature is not supported in Token Ring VLANs.
- Layer 2 traceroute opens a listening socket on the User Datagram Protocol (UDP) port 2228 that can be accessed remotely with any IPv4 address, and does not require any authentication. This UDP socket allows to read VLAN information, links, presence of particular MAC addresses, and CDP neighbor information, from the device. This information can be used to eventually build a complete picture of the Layer 2 network topology.
- Layer 2 traceroute is enabled by default and can be disabled by running the **no l2 traceroute** command in global configuration mode. To re-enable Layer 2 traceroute, use the **l2 traceroute** command in global configuration mode.

IP Traceroute

You can use IP traceroute to identify the path that packets take through the network on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as routers, that the traffic passes through on the way to the destination.

Your Device can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the Device is the destination of the traceroute, it is displayed as the final destination in the traceroute output. Intermediate Device do not show up in the traceroute output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate Device is a multilayer Device that is routing a particular packet, this Device shows up as a hop in the traceroute output.

The **traceroute** privileged EXEC command uses the Time To Live (TTL) field in the IP header to cause routers and servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends an Internet Control Message Protocol (ICMP) time-to-live-exceeded message to the sender. Traceroute finds the address of the first hop by examining the source address field of the ICMP time-to-live-exceeded message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the time-to-live-exceeded message to the source. This process continues until the

TTL is incremented to a value large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To learn when a datagram reaches its destination, traceroute sets the UDP destination port number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP *port-unreachable* error to the source. Because all errors except port-unreachable errors come from intermediate hops, the receipt of a port-unreachable error means that this message was sent by the destination port.

Time Domain Reflector Guidelines

You can use the Time Domain Reflector (TDR) feature to diagnose and resolve cabling problems. When running TDR, a local device sends a signal through a cable and compares the reflected signal to the initial signal.

TDR is supported only on 10/100/1000 copper Ethernet ports. It is not supported on 10-Gigabit Ethernet ports and on SFP module ports.

TDR can detect these cabling problems:

- Open, broken, or cut twisted-pair wires—The wires are not connected to the wires from the remote device.
- Shorted twisted-pair wires—The wires are touching each other or the wires from the remote device. For example, a shorted twisted pair can occur if one wire of the twisted pair is soldered to the other wire.

If one of the twisted-pair wires is open, TDR can find the length at which the wire is open.

Use TDR to diagnose and resolve cabling problems in these situations:

- Replacing a Device
- Setting up a wiring closet
- Troubleshooting a connection between two devices when a link cannot be established or when it is not operating properly

When you run TDR, the Device reports accurate information in these situations:

- The cable for the gigabit link is a solid-core cable.
- The open-ended cable is not terminated.

When you run TDR, the Device does not report accurate information in these situations:

- The cable for the gigabit link is a twisted-pair cable or is in series with a solid-core cable.
- The link is a 10-megabit or a 100-megabit link.
- The cable is a stranded cable.
- The link partner is a Cisco IP Phone.
- The link partner is not IEEE 802.3 compliant.

Debug Commands



Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. It is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments.

Onboard Failure Logging on the Switch

You can use the onboard failure logging (OBFL) feature to collect information about the Device. The information includes uptime, temperature, and voltage information and helps Cisco technical support representatives to troubleshoot Device problems. We recommend that you keep OBFL enabled and do not erase the data stored in the flash memory.

By default, OBFL is enabled. It collects information about the Device and small form-factor pluggable (SFP) modules. The Device stores this information in the flash memory:

- CLI commands—Record of the OBFL CLI commands that are entered on a standalone Device.
- Environment data—Unique device identifier (UDI) information for a standalone Device and for all the connected FRU devices: the product identification (PID), the version identification (VID), and the serial number.
- Message—Record of the hardware-related system messages generated by a standalone Device .
- Power over Ethernet (PoE)—Record of the power consumption of PoE ports on a standalone Device .
- Temperature—Temperature of a standalone Device .
- Uptime data—Time when a standalone Device starts, the reason the Device restarts, and the length of time the Device has been running since it last restarted.
- Voltage—System voltages of a standalone Device .

You should manually set the system clock or configure it by using Network Time Protocol (NTP).

When the Device is running, you can retrieve the OBFL data by using the **show logging onboard** privileged EXEC commands. If the Device fails, contact your Cisco technical support representative to find out how to retrieve the data.

When an OBFL-enabled Device is restarted, there is a 10-minute delay before logging of new data begins.

Possible Symptoms of High CPU Utilization

Excessive CPU utilization might result in these symptoms, but the symptoms might also result from other causes:



Note You may see increased system memory usage when Cisco Catalyst 4500E Supervisor Engine 8-E is used in wireless mode.

- Spanning tree topology changes
- EtherChannel links brought down due to loss of communication
- Failure to respond to management requests (ICMP ping, SNMP timeouts, slow Telnet or SSH sessions)
- UDLD flapping
- IP SLAs failures because of SLAs responses beyond an acceptable threshold
- DHCP or IEEE 802.1x failures if the switch does not forward or respond to requests

Layer 3 switches:

- Dropped packets or increased latency for packets routed in software
- BGP or OSPF routing topology changes
- HSRP flapping

How to Troubleshoot the Software Configuration

Recovering from a Software Failure

Switch software can be corrupted during an upgrade by downloading the wrong file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

This procedure uses the Xmodem Protocol to recover from a corrupt or wrong image file. There are many software packages that support the Xmodem Protocol, and this procedure is largely dependent on the emulation software that you are using.

This recovery procedure requires that you have physical access to the switch.

Step 1 From your PC, download the software image tar file (*image_filename.tar*) from Cisco.com. The Cisco IOS image is stored as a bin file in a directory in the tar file. For information about locating the software image files on Cisco.com, see the release notes.

Step 2 Extract the bin file from the tar file. If you are using Windows, use a zip program that can read a tar file. Use the zip program to navigate. If you are using Windows, use a zip program that can read a tar file. Use the zip program to navigate. If you are using UNIX, follow these steps:

- a) Display the contents of the tar file by using the **tar -tvf <image_filename.tar>** UNIX command.

Example:

```
unix-1% tar -tvf image_filename.tar
```

- b) Locate the bin file, and extract it by using the `tar -xvf <image_filename.tar> <image_filename.bin>` UNIX command.

Example:

```
unix-1% tar -xvf image_filename.tar image_filename.bin
x c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin, 2928176 bytes, 5720
tape blocks
```

- c) Verify that the bin file was extracted by using the `ls -l <image_filename.bin>` UNIX command.

Example:

```
unix-1% ls -l image_filename.bin
-rw-r--r--  1 boba      2928176 Apr 21 12:01
c2960x-universalk9-mz.150-2.0.66.UCP/c2960x-universalk9-mz.150-2.0.66.UCP.bin
```

- Step 3** Connect your PC with terminal-emulation software supporting the Xmodem Protocol to the switch console port.
- Step 4** Set the line speed on the emulation software to 9600 baud.
- Step 5** Unplug the switch power cord.
- Step 6** Press the **Mode** button, and at the same time reconnect the power cord to the switch. You can release the Mode button a second or two after the LED above port 1 goes off. Several lines of information about the software appear along with instructions.

Example:

```
The system has been interrupted prior to initializing the flash file system. The following
commands will initialize the flash file system, and finish loading the operating system
software#
```

```
flash_init
load_helper
boot
```

- Step 7** Initialize the flash file system.

Example:

```
switch: flash_init
```

- Step 8** If you had set the console port speed to any speed other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

- Step 9** Load any helper files.

Example:

```
switch: load_helper
```

- Step 10** Start the file transfer by using the Xmodem Protocol.

Example:

```
switch: copy xmodem: flash:image_filename.bin
```

- Step 11** After the Xmodem request appears, use the appropriate command on the terminal-emulation software to start the transfer and to copy the software image into flash memory.
- Step 12** Boot the newly downloaded Cisco IOS image.
- Example:**
- ```
switch: boot flash:image_filename.bin
```
- Step 13** Use the **archive download-sw** privileged EXEC command to download the software image to the switch or to the switch stack.
- Step 14** Use the **reload** privileged EXEC command to restart the switch and to verify that the new software image is operating properly.
- Step 15** Delete the **flash:image\_filename.bin** file from the switch.
- 

## Recovering from a Lost or Forgotten Password

The default configuration for the switch allows an end user with physical access to the switch to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the switch.



**Note** On these switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message shows this during the recovery process.

---

You enable or disable password recovery by using the **service password-recovery** global configuration command.

---

- Step 1** Connect a terminal or PC to the switch.
- Connect a terminal or a PC with terminal-emulation software to the switch console port.
- Or
- Connect a PC to the Ethernet management port.
- Step 2** Set the line speed on the emulation software to 9600 baud.
- Step 3** On a switch, power off the switch.
- Step 4** Reconnect the power cord to the switch. Within 15 seconds, press the **Mode** button while the System LED is still flashing green. Continue pressing the **Mode** button until all the system LEDs turn on and remain solid, then release the **Mode** button.

Several lines of information about the software appear with instructions, informing you if the password recovery procedure has been disabled or not.

- If you see a message that begins with this statement:

The system has been interrupted prior to initializing the flash file system. The following commands will initialize the flash file system

proceed to the "Procedure with Password Recovery Enabled" section, and follow the steps.

- If you see a message that begins with this statement:

The password-recovery mechanism has been triggered, but is currently disabled.

proceed to the "Procedure with Password Recovery Disabled" section, and follow the steps.

**Step 5** After recovering the password, reload the switch.

On a switch:

```
Switch> reload
Proceed with reload? [confirm] y
```

---

## Procedure with Password Recovery Enabled

If the password-recovery operation is enabled, this message appears:

The system has been interrupted prior to initializing the flash file system. The following commands will initialize the flash file system, and finish loading the operating system software:

```
flash_init
load_helper
boot
```

---

**Step 1** Initialize the flash file system.

Switch: **flash\_init**

**Step 2** If you had set the console port speed to any number other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

**Step 3** Load any helper files.

Switch: **load\_helper**

**Step 4** Display the contents of flash memory.

```
Switch: dir: flash:
Directory of flash:
 13 drwx 192 Mar 01 2013 22:30:48
c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin
 11 -rwx 5825 Mar 01 2013 22:31:59 config.text

16128000 bytes total (10003456 bytes free)
```

**Step 5** Rename the configuration file to config.text.old



This file contains the password definition.

```
Switch: rename flash: config.text flash: config.text.old
```

**Step 6** Boot up the system.

```
Switch: boot
```

You are prompted to start the setup program. Enter **N** at the prompt.

```
Continue with the configuration dialog?? [yes/no]: No
```

**Step 7** At the switch prompt, enter privileged EXEC mode.

```
Switch> enable
Switch#
```

**Step 8** Rename the configuration file to its original name.

```
Switch# rename flash: config.text.old flash: config.text
```

**Note** Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized. Failure to follow this step can result in a lost configuration depending on how your device is set up.

**Step 9** Copy the configuration file into memory

```
Switch# copy flash: config.text system: running-config
Source filename [config.text]?
Destination filename [running-config]?
```

Press **Return** in response to the confirmation prompts. The configuration file is now reloaded, and you can change the password.

**Step 10** Enter global configuration mode.

```
Switch# configure terminal
```

**Step 11** Change the password.

```
Switch(config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

**Step 12** Return to privileged EXEC mode.

```
Switch(config)# exit
Switch#
```

**Step 13** Write the running configuration to the startup configuration file.

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

**Note** This procedure is likely to leave your switch virtual interface in a shutdown state. You can see which interface is in this state by entering the **show running-config** privileged EXEC command. To reenabling the interface, enter the **interface vlan *vlan-id*** global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

**Step 14** Boot the device with the *packages.conf* file from flash.

```
Switch: boot flash:packages.conf
```

**Step 15** Reload the switch stack.

```
Switch# reload
```

---

## Procedure with Password Recovery Disabled

If the password-recovery mechanism is disabled, this message appears:

```
The password-recovery mechanism has been triggered, but
is currently disabled. Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point. However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.
```

```
Would you like to reset the system back to the default configuration (y/n)?
```




---

**Caution** Returning the Device to the default configuration results in the loss of all existing configurations. We recommend that you contact your system administrator to verify if there are backup Device and VLAN configuration files.

---

- If you enter **n** (no), the normal boot process continues as if the **Mode** button had not been pressed; you cannot access the boot loader prompt, and you cannot enter a new password. You see the message:

```
Press Enter to continue.....
```

- If you enter **y** (yes), the configuration file in flash memory and the VLAN database file are deleted. When the default configuration loads, you can reset the password.
- 

**Step 1** Choose to continue with password recovery and delete the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? y
```

**Step 2** Display the contents of flash memory:

```
Switch: dir flash:
```

The Device file system appears.

**Step 3** Boot up the system:

```
Switch: boot
```

You are prompted to start the setup program. To continue with password recovery, enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

**Step 4** At the Device prompt, enter privileged EXEC mode:

```
Switch> enable
```

**Step 5** Enter global configuration mode:

```
Switch# configure terminal
```

**Step 6** Change the password:

```
Switch(config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

**Step 7** Return to privileged EXEC mode:

```
Switch(config)# exit
Switch#
```

**Note** Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized.

**Step 8** Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

**Step 9** You must now reconfigure the Device. If the system administrator has the backup Device and VLAN configuration files available, you should use those.

---

## Recovering from a Command Switch Failure

This section describes how to recover from a failed command switch. You can configure a redundant command switch group by using the Hot Standby Router Protocol (HSRP).



**Note** This feature is introduced from Cisco IOS Release 15.2(5)E2.

If you have not configured a standby command switch, and your command switch loses power or fails in some other way, management contact with the member switches is lost, and you must install a new command switch. However, connectivity between switches that are still connected is not affected, and the member switches forward packets as usual. You can manage the members as standalone switches through the console port, or, if they have IP addresses, through the other management interfaces.

You can prepare for a command switch failure by assigning an IP address to a member switch or another switch that is command-capable, making a note of the command-switch password, and cabling your cluster to provide redundant connectivity between the member switches and the replacement command switch. These sections describe two solutions for replacing a failed command switch:

- Replacing a Failed Command Switch with a Cluster Member
- Replacing a Failed Command Switch with Another Switch

These recovery procedures require that you have physical access to the switch. For information on command-capable switches, see the release notes.

## Replacing a Failed Command Switch with a Cluster Member

To replace a failed command switch with a command-capable member in the same cluster, follow these steps

- 
- Step 1** Disconnect the command switch from the member switches, and physically remove it from the cluster.
- Step 2** Insert the member switch in place of the failed command switch, and duplicate its connections to the cluster members.
- Step 3** Start a CLI session on the new command switch.
- You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see *Catalyst 3560-CX and 2960-CX Switch Hardware Installation Guide*
- Step 4** At the switch prompt, enter privileged EXEC mode.
- Example:**
- ```
Switch> enable
Switch#
```
- Step 5** Enter the password of the *failed command switch*.
- Step 6** Enter global configuration mode.
- Example:**
- ```
Switch# configure terminal
```
- Enter configuration commands, one per line. End with CNTL/Z.
- Step 7** Remove the member switch from the cluster.
- Example:**
- ```
Switch(config)# no cluster commander-address
```
- Step 8** Return to privileged EXEC mode.

Example:

```
Switch(config)# end
Switch#
```

- Step 9** Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter EXEC mode, enter **setup**, and press **Return**.

Example:

```
Switch# setup

--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

- Step 10** Enter **Y** at the first prompt.

Example:

```
The prompts in the setup program vary depending on the member switch that you selected to be the
command switch:
Continue with configuration dialog? [yes/no]: y

or

Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

- Step 11** Respond to the questions in the setup program.

When prompted for the hostname, it is limited to 28 characters and 31 characters on a member switch. Do not use *-n*, where *n* is a number, as the last characters in a hostname for any switch. When prompted for the Telnet (virtual terminal) password, it is 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

- Step 12** When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

- Step 13** When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.

- Step 14** When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

- Step 15** After the initial configuration displays, verify that the addresses are correct.

- Step 16** If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

- Step 17** Start your browser, and enter the IP address of the new command switch.

- Step 18** From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

Replacing a Failed Command Switch with Another Switch

To replace a failed command switch with a switch that is command-capable but not part of the cluster, follow these steps:

- Step 1** Insert the new switch in place of the failed command switch, and duplicate its connections to the cluster members.
- Step 2** You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see the switch hardware installation guide.
- Step 3** At the switch prompt, enter privileged EXEC mode.

Example:

```
Switch> enable
Switch#
```

- Step 4** Enter the password of the *failed command switch*.
- Step 5** Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter EXEC mode, enter **setup**, and press **Return**.

Example:

```
Switch# setup

--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

- Step 6** Enter **Y** at the first prompt.

Example:

The prompts in the setup program vary depending on the member switch that you selected to be the command switch:

```
Continue with configuration dialog? [yes/no]: y
```

or

```
Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

- Step 7** Respond to the questions in the setup program.
- When prompted for the hostname, it is limited to 28 characters and 31 characters on a member switch. Do not use *-n*, where *n* is a number, as the last characters in a hostname for any switch. When prompted for the Telnet (virtual terminal) password, it is 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.
- Step 8** When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.
- Step 9** When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.
- Step 10** When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

Step 11 After the initial configuration displays, verify that the addresses are correct.

Step 12 If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

Step 13 Start your browser, and enter the IP address of the new command switch.

Step 14 From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

Preventing Autonegotiation Mismatches

The IEEE 802.3ab autonegotiation protocol manages the switch settings for speed (10 Mb/s, 100 Mb/s, and 1000 Mb/s, excluding SFP module ports) and duplex (half or full). There are situations when this protocol can incorrectly align these settings, reducing performance. A mismatch occurs under these circumstances:

- A manually set speed or duplex parameter is different from the manually set speed or duplex parameter on the connected port.
- A port is set to autonegotiate, and the connected port is set to full duplex with no autonegotiation.

To maximize switch performance and ensure a link, follow one of these guidelines when changing the settings for duplex and speed:

- Let both ports autonegotiate both speed and duplex.
- Manually set the speed and duplex parameters for the ports on both ends of the connection.



Note If a remote device does not autonegotiate, configure the duplex settings on the two ports to match. The speed parameter can adjust itself even if the connected port does not autonegotiate.

Troubleshooting SFP Module Security and Identification

Cisco small form-factor pluggable (SFP) modules have a serial EEPROM that contains the module serial number, the vendor name and ID, a unique security code, and cyclic redundancy check (CRC). When an SFP module is inserted in the Device, the Device software reads the EEPROM to verify the serial number, vendor name and vendor ID, and recompute the security code and CRC. If the serial number, the vendor name or vendor ID, the security code, or CRC is invalid, the software generates a security error message and places the interface in an error-disabled state.



Note The security error message references the GBIC_SECURITY facility. The Device supports SFP modules and does not support GBIC modules. Although the error message text refers to GBIC interfaces and modules, the security messages actually refer to the SFP modules and module interfaces.

If you are using a non-Cisco SFP module, remove the SFP module from the Device, and replace it with a Cisco module. After inserting a Cisco SFP module, use the **errdisable recovery cause gbic-invalid** global

configuration command to verify the port status, and enter a time interval for recovering from the error-disabled state. After the elapsed interval, the Device brings the interface out of the error-disabled state and retries the operation. For more information about the **errdisable recovery** command, see the command reference for this release.

If the module is identified as a Cisco SFP module, but the system is unable to read vendor-data information to verify its accuracy, an SFP module error message is generated. In this case, you should remove and reinsert the SFP module. If it continues to fail, the SFP module might be defective.

Monitoring SFP Module Status

You can check the physical or operational status of an SFP module by using the **show interfaces transceiver** privileged EXEC command. This command shows the operational status, such as the temperature and the current for an SFP module on a specific interface and the alarm status. You can also use the command to check the speed and the duplex settings on an SFP module. For more information, see the **show interfaces transceiver** command in the command reference for this release.

Executing Ping

If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or have IP routing configured to route between those subnets.

IP routing is disabled by default on all Device.



Note Though other protocol keywords are available with the **ping** command, they are not supported in this release.

Use this command to ping another device on the network from the Device:

Command	Purpose
ping ip <i>host address</i>	Pings a remote host through IP or by supplying the hostname or network address.
Switch# ping 172.20.52.3	

Monitoring Temperature

The Device monitors the temperature conditions and uses the temperature information to control the fans.

Use the **show env temperature status** privileged EXEC command to display the temperature value, state, and thresholds. The temperature value is the temperature in the Device (not the external temperature). You can configure only the yellow threshold level (in Celsius) by using the **system env temperature threshold yellow** *value* global configuration command to set the difference between the yellow and red thresholds. You cannot configure the green or red thresholds. For more information, see the command reference for this release.

Monitoring the Physical Path

You can monitor the physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

Table 168: Monitoring the Physical Path

Command	Purpose
tracetroute mac [interface <i>interface-id</i>] { <i>source-mac-address</i> } [interface <i>interface-id</i>] { <i>destination-mac-address</i> } [vlan <i>vlan-id</i>] [detail]	Displays the Layer 2 path taken by the packets from the specified source MAC address to the specified destination MAC address.
tracetroute mac ip { <i>source-ip-address</i> <i>source-hostname</i> } { <i>destination-ip-address</i> <i>destination-hostname</i> } [detail]	Displays the Layer 2 path taken by the packets from the specified source IP address or hostname to the specified destination IP address or hostname.

Executing IP Traceroute



Note Though other protocol keywords are available with the **tracetroute** privileged EXEC command, they are not supported in this release.

Command	Purpose
traceroute ip <i>host</i> Switch# traceroute ip 192.51.100.1	Traces the path that packets take through the network.

Running TDR and Displaying the Results

To run TDR, enter the **test cable-diagnostics tdr interface** *interface-id* privileged EXEC command.

To display the results, enter the **show cable-diagnostics tdr interface** *interface-id* privileged EXEC command.

Redirecting Debug and Error Message Output

By default, the network server sends the output from **debug** commands and system error messages to the console. If you use this default, you can use a virtual terminal connection to monitor debug output instead of connecting to the console port or the Ethernet management port.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.



Note Be aware that the debugging destination you use affects system overhead. When you log messages to the console, very high overhead occurs. When you log messages to a virtual terminal, less overhead occurs. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

For more information about system message logging, see *Configuring System Message Logging*.

Using the show platform forward Command

The output from the **show platform forward** privileged EXEC command provides some useful information about the forwarding results if a packet entering an interface is sent through the system. Depending upon the parameters entered about the packet, the output provides lookup table results and port maps used to calculate forwarding destinations, bitmaps, and egress information.

Most of the information in the output from the command is useful mainly for technical support personnel, who have access to detailed information about the Device application-specific integrated circuits (ASICs). However, packet forwarding information can also be helpful in troubleshooting.

Configuring OBFL



Caution We recommend that you do not disable OBFL and that you do not remove the data stored in the flash memory.

- To enable OBFL, use the **hw-switch switch** [*switch-number*] **logging onboard** [**message level** *level*] global configuration command. On switches, the range for *switch-number* is from 1 to 9. Use the **message level** *level* parameter to specify the severity of the hardware-related messages that the switch generates and stores in the flash memory.
- To copy the OBFL data to the local network or a specific file system, use the **copy onboard switch** *switch-number* **url** *url-destination* privileged EXEC command.
- To disable OBFL, use the **no hw-switch switch** [*switch-number*] **logging onboard** [**message level**] global configuration command.
- To clear all the OBFL data in the flash memory except for the uptime and CLI command information, use the **clear onboard switch** *switch-number* privileged EXEC command.
- You can enable or disable OBFL on a member switch from the active stack.

For more information about the commands in this section, see the command reference for this release.

Verifying Troubleshooting of the Software Configuration

Displaying OBFL Information

Table 169: Commands for Displaying OBFL Information

Command	Purpose
show logging onboard [module[switch-number]] cliilog Switch# show logging onboard 1 cliilog	Displays the OBFL CLI commands that were entered on a standalone switch.
show logging onboard [module[switch-number]] environment Switch# show logging onboard 1 environment	Displays the UDI information for a standalone switch and for all the connected FRU devices: the PID, the VID, and the serial number.
show logging onboard [module[switch-number]] message Switch# show logging onboard 1 message	Displays the hardware-related messages generated by a standalone switch.
show logging onboard [module[switch-number]] poe Switch# show logging onboard 1 poe	Displays the power consumption of PoE ports on a standalone switch.
show logging onboard [module[switch-number]] temperature Switch# show logging onboard 1 temperature	Displays the temperature of a standalone switch or.
show logging onboard [module[switch-number]] uptime Switch# show logging onboard 1 uptime	Displays the time when a standalone switch or the specified stack members start, the reason the standalone switch or specified stack members restart, and the length of time that the standalone switch have been running since they last restarted.
show logging onboard [module[switch-number]] voltage Switch# show logging onboard 1 voltage	Displays the system voltages of a standalone switch.
show logging onboard [module[switch-number]] continuous Switch# show logging onboard 1 continuous	Displays the data in the continuous file.

Command	Purpose
show logging onboard [module[switch-number]] detail Switch# show logging onboard 1 detail	Displays both the continuous and summary data .
show logging onboard [module[switch-number]] endhh:mm:ss Switch# show logging onboard 1 end 13:00:15 jul 2013	Displays end time and date on a standalone switch.
show logging onboard [module[switch-number]] Switch# show logging onboard 1	Displays OBFL information about the specified switches in the system.
show logging onboard [module[switch-number]] raw Switch# show logging onboard 1 raw	Displays the raw information on a standalone switch.
show logging onboard [module[switch-number]] start Switch# show logging onboard 1 start 13:00:10 jul 2013	Displays the start time and date on a standalone switch.
show logging onboard [module[switch-number]] status Switch# show logging onboard 1 status	Displays status information on a standalone switch.
show logging onboard [module[switch-number]] summary Switch# show logging onboard 1 summary	Displays both the data in the summary file

Example: Verifying the Problem and Cause for High CPU Utilization

To determine if high CPU utilization is a problem, enter the **show processes cpu sorted** privileged EXEC command. Note the underlined information in the first line of the output example.

```
Switch# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<output truncated>
```

This example shows normal CPU utilization. The output shows that utilization for the last 5 seconds is 8%/0%, which has this meaning:

- The total CPU utilization is 8 percent, including both time running Cisco IOS processes and time spent handling interrupts.
- The time spent handling interrupts is zero percent.

Table 170: Troubleshooting CPU Utilization Problems

Type of Problem	Cause	Corrective Action
Interrupt percentage value is almost as high as total CPU utilization value.	The CPU is receiving too many packets from the network.	Determine the source of the network packet. Stop the flow, or change the switch configuration. See the section on “Analyzing Network Traffic.”
Total CPU utilization is greater than 50% with minimal time spent on interrupts.	One or more Cisco IOS process is consuming too much CPU time. This is usually triggered by an event that activated the process.	Identify the unusual event, and troubleshoot the root cause. See the section on “Debugging Active Processes.”

Scenarios for Troubleshooting the Software Configuration

Scenarios to Troubleshoot Power over Ethernet (PoE)

Table 171: Power over Ethernet Troubleshooting Scenarios

Symptom or Problem	Possible Cause and Solution
<p>Only one port does not have PoE.</p> <p>Trouble is on only one switch port. PoE and non-PoE devices do not work on this port, but do on other ports.</p>	<p>Verify that the powered device works on another PoE port.</p> <p>Use the show run, or show interface status user EXEC commands to verify that the port is not shut down or error-disabled.</p> <p>Note Most switches turn off port power when the port is shut down, even though the IEEE specifications make this optional.</p> <p>Verify that power inline never is not configured on that interface or port.</p> <p>Verify that the Ethernet cable from the powered device to the switch port is good: Connect a known good non-PoE Ethernet device to the Ethernet cable, and make sure that the powered device establishes a link and exchanges traffic with another host.</p> <p>Note Cisco powered device works only with straight cable and not with crossover one.</p> <p>Verify that the total cable length from the switch front panel to the powered device is not more than 100 meters.</p> <p>Disconnect the Ethernet cable from the switch port. Use a short Ethernet cable to connect a known good Ethernet device directly to this port on the switch front panel (not on a patch panel). Verify that it can establish an Ethernet link and exchange traffic with another host, or ping the port VLAN SVI. Next, connect a powered device to this port, and verify that it powers on.</p> <p>If a powered device does not power on when connected with a patch cord to the switch port, compare the total number of connected powered devices to the switch power budget (available PoE). Use the show power inline command to verify the amount of available power.</p>

Symptom or Problem	Possible Cause and Solution
<p>No PoE on all ports or a group of ports. Trouble is on all switch ports. Nonpowered Ethernet devices cannot establish an Ethernet link on any port, and PoE devices do not power on.</p>	<p>If there is a continuous, intermittent, or reoccurring alarm related to power, replace the power supply if possible it is a field-replaceable unit. Otherwise, replace the switch.</p> <p>If the problem is on a consecutive group of ports but not all ports, the power supply is probably not defective, and the problem could be related to PoE regulators in the switch.</p> <p>Use the show log privileged EXEC command to review alarms or system messages that previously reported PoE conditions or status changes.</p> <p>If there are no alarms, use the show interface status command to verify that the ports are not shut down or error-disabled. If ports are error-disabled, use the shut and no shut interface configuration commands to reenable the ports.</p> <p>Use the show env power and show power inline privileged EXEC commands to review the PoE status and power budget (available PoE).</p> <p>Review the running configuration to verify that power inline never is not configured on the ports.</p> <p>Connect a nonpowered Ethernet device directly to a switch port. Use only a short patch cord. Do not use the existing distribution cables. Enter the shut and no shut interface configuration commands, and verify that an Ethernet link is established. If this connection is good, use a short patch cord to connect a powered device to this port and verify that it powers on. If the device powers on, verify that all intermediate patch panels are correctly connected.</p> <p>Disconnect all but one of the Ethernet cables from switch ports. Using a short patch cord, connect a powered device to only one PoE port. Verify the powered device does not require more power than can be delivered by the switch port.</p> <p>Use the show power inline privileged EXEC command to verify that the powered device can receive power when the port is not shut down. Alternatively, watch the powered device to verify that it powers on.</p> <p>If a powered device can power on when only one powered device is connected to the switch, enter the shut and no shut interface configuration commands on the remaining ports, and then reconnect the Ethernet cables one at a time to the switch PoE ports. Use the show interface status and show power inline privileged EXEC commands to monitor inline power statistics and port status.</p> <p>If there is still no PoE at any port, a fuse might be open in the PoE section of the power supply. This normally produces an alarm. Check the log again for alarms reported earlier by system messages.</p>

Symptom or Problem	Possible Cause and Solution
<p>Cisco pre-standard powered device disconnects or resets.</p> <p>After working normally, a Cisco phone intermittently reloads or disconnects from PoE.</p>	<p>Verify all electrical connections from the switch to the powered device. Any unreliable connection results in power interruptions and irregular powered device functioning such as erratic powered device disconnects and reloads.</p> <p>Verify that the cable length is not more than 100 meters from the switch port to the powered device.</p> <p>Notice what changes in the electrical environment at the switch location or what happens at the powered device when the disconnect occurs.</p> <p>Notice whether any error messages appear at the same time a disconnect occurs. Use the show log privileged EXEC command to review error messages.</p> <p>Verify that an IP phone is not losing access to the Call Manager immediately before the reload occurs. (It might be a network problem and not a PoE problem.)</p> <p>Replace the powered device with a non-PoE device, and verify that the device works correctly. If a non-PoE device has link problems or a high error rate, the problem might be an unreliable cable connection between the switch port and the powered device.</p>
<p>IEEE 802.3af-compliant or IEEE 802.3at-compliant powered devices do not work on Cisco PoE switch.</p> <p>A non-Cisco powered device is connected to a Cisco PoE switch, but never powers on or powers on and then quickly powers off. Non-PoE devices work normally.</p>	<p>Use the show power inline command to verify that the switch power budget (available PoE) is not depleted before or after the powered device is connected. Verify that sufficient power is available for the powered device type before you connect it.</p> <p>Use the show interface status command to verify that the switch detects the connected powered device.</p> <p>Use the show log command to review system messages that reported an overcurrent condition on the port. Identify the symptom precisely: Does the powered device initially power on, but then disconnect? If so, the problem might be an initial surge-in (or <i>inrush</i>) current that exceeds a current-limit threshold for the port.</p>

Configuration Examples for Troubleshooting Software

Example: Pinging an IP Host

This example shows how to ping an IP host:

```
Switch# ping 172.20.52.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!
```



```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Switch#
```

Table 172: Ping Output Display Characters

Character	Description
!	Each exclamation point means receipt of a reply.
.	Each period means the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

To end a ping session, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

Example: Performing a Traceroute to an IP Host

This example shows how to perform a **traceroute** to an IP host:

```
Switch# traceroute ip 192.0.2.10

Type escape sequence to abort.
Tracing the route to 192.0.2.10

 1 192.0.2.1 0 msec 0 msec 4 msec
 2 192.0.2.203 12 msec 8 msec 0 msec
 3 192.0.2.100 4 msec 0 msec 0 msec
 4 192.0.2.10 0 msec 4 msec 0 msec
```

The display shows the hop count, the IP address of the router, and the round-trip time in milliseconds for each of the three probes that are sent.

Table 173: Traceroute Output Display Characters

Character	Description
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable. Usually, this output means that an access list is blocking traffic.
H	Host unreachable.
N	Network unreachable.

Character	Description
P	Protocol unreachable.
Q	Source quench.
U	Port unreachable.

To end a trace in progress, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

Example: Enabling All System Diagnostics



Caution Because debugging output takes priority over other network traffic, and because the **debug all** privileged EXEC command generates more output than any other **debug** command, it can severely diminish switch performance or even render it unusable. In virtually all cases, it is best to use more specific **debug** commands.

This command disables all-system diagnostics:

```
Switch# debug all
```

The **no debug all** privileged EXEC command disables all diagnostic output. Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands enabled.



CHAPTER 86

Information About Licensing

- [Restrictions for Configuring Licenses, on page 1723](#)
- [Information About Licensing, on page 1723](#)
- [How to Configure Add-On License Levels, on page 1726](#)
- [Configuration Examples for License Levels, on page 1729](#)
- [Feature History for Information About Licensing, on page 1731](#)

Restrictions for Configuring Licenses

- Members of a switch stack must run the same license level (base license level and add-on). If the license level is different with a mismatched base license, the switch will not join the stack until it is changed and rebooted from the active stack. Mismatched add-on licenses are automatically synced by the active stack.
- A permanent license can be moved from one device to another. To activate a license, you must reboot your switch.
- An expired evaluation license cannot be reactivated after reboot.

Information About Licensing

Overview of License Levels

Software features on the switch are available with base (also known as feature sets) and add-on license levels. Their validity duration determines the license type.

- **Base license levels** for a switch are indicated by the switch model number. They are always permanent licenses, without an expiration date.
- **Add-on license levels** provide Cisco innovations on the switch, as well as on the Cisco Digital Network Architecture Center (Cisco DNA Center). Add-on licenses may be ordered only with a term license type, for a three, five, or seven year period.

Base Licenses

Cisco Catalyst 3560-CX Series compact switches ship with the IP Base license and can be upgraded to use the IP Services license. Cisco Catalyst 2960-CX Series compact switches ship with the LAN Base license level.



Note On Cisco Catalyst 2960-CX Series, the base license level is bound to the hardware model and cannot be changed.

Add-On Licenses

On Cisco Catalyst 3560-CX Series compact switches, the following add-on licenses are available:

- DNA Essentials
- DNA Advantage

On Cisco Catalyst 2960-CX Series compact switches, the DNA Essentials add-on license is available

The following guidelines apply to Add-on Licenses:

- A Reboot is not required when you configure an add-on license.
- Add-on licenses may be ordered for a three, five, or seven year period.
- You must set up Cisco SSM to receive daily e-mail alerts, to be notified of expiring add-on licenses that you want to renew.
- For the Cisco Catalyst 3560-CX Series: When you order the IP Base + DNA Advantage combination, only three virtual networks are available with SDA.
- For the Cisco Catalyst 2960-CX Series: Only the DNA Essentials add-on license is available. (Although visible on the CLI, the DNA Advantage license level is not available).

License States

You can also access the license information by using the **show license** command in the privileged EXEC mode.

Table 174: Right-to-use license states

License State	Description
Active, In Use	EULA was accepted and the license is in use after device reboot.
Active, Not In Use	EULA was accepted and the switch is ready to use when the license is enabled.
Not Activated	EULA was not accepted.

The following example shows how to display the license level of the switch. The example shows LAN Base as the active license and as the one that is in use.

```
Switch# show license

Index 1
License Name      : lanlite
Period left       : 0 minute 0 second
License Type: Permanent
License State: Inactive
Index 2
License Name      : lanbase
Period left       : 0 minute 0 second
License Type: Permanent
License State: Active, In use

Index 3
License Name      : dna-essentials
Period left       : CSSM Managed
License Type      : Subscription
License State     : Active, In use

Index 4
License Name      : dna-advantage
Period left       : CSSM Managed
License Type      : Subscription
License State     : Not Activated
```

Guidelines to follow when monitoring your image based license state:

- A purchased permanent license is set to Active, In Use state only after a switch reboot.
- If more than one license was purchased, a reboot will activate the license with the highest feature set. For instance, the LAN Base license is activated and not the LAN Lite license.
- The remaining licenses purchased after switch reboot, stay in Active, Not In Use state.

Guidelines for License Types

Licenses may be of the permanent or term type only.

- Permanent: For a license level, and without an expiration date. The basic license type for the switch is determined by the model and is always permanent.
- Term: For a license level, and for a three, five, or seven year period. Add-on licenses (DNA Essentials and DNA Advantage) may be ordered only with a term license type.

Ordering with Smart Accounts

We recommend that you use Smart Accounts to order devices as well as licenses. Smart Accounts enable you to manage all of your software licenses for switches, routers, firewalls, access-points or tools from one centralized website. To create Smart Accounts, use the Cisco Smart Software Manager (Cisco SSM).



Note This is especially relevant to the term licenses that you order, because information about the expiry of term licenses is available only through the Cisco SSM website.

For more information about Cisco SSM, see: <http://www.cisco.com/c/en/us/buy/smart-accounts/software-licensing.html>

License Activation for Switch Stacks

LAN Base models can stack with LAN Base models only.

The active stack is activated with a license from its active console. The license level for members in the stack can be activated at the same time.

To change the license level, do not disconnect a newly added stack member if the stack cables are connected. Instead, use the active console to set the new member's license level at the same license level as an active stack and reboot the new member to join the stack.

Reboot is required only for the base license; not when you configure an add-on license

How to Configure Add-On License Levels

The following sections provide information on how to configure Add-on License Levels.

Activating an Image Based Add-on License

The following steps can be used to activate an image based license.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **license boot level addon *addon-license***
4. **license accept end user agreement force**
5. **show license right-to-use usage**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	license boot level addon <i>addon-license</i> Example: Device(config)# <code>license boot level addon dna-essentials</code>	Specifies the add-on license level. The following options are available: <ul style="list-style-type: none"> • DNA Essentials • DNA Advantage
Step 4	license accept end user agreement force Example: Device(config)# <code>license accept end user agreement force</code>	Enables acceptance of the end-user license agreement (EULA). Note To configure an add-on license EULA acceptance is not mandatory, but you will not be able to use or configure the DNAC features until you complete this step.
Step 5	show license right-to-use usage Example: Device(config)# <code>show license right-to-use usage</code>	Displays detailed usage information. Other options are available with the show license right-to-use command .

Activating an Image Based Add-on License on Cisco Catalyst 3560-CX Series

The following steps can be used to activate an image based license.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `license right-to-use [activate | deactivate] [addon { dna-essentials | dna-advantage }] { subscription | evaluation } [acceptEULA]`
4. `show license right-to-use usage`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>license right-to-use [activate deactivate] [addon { dna-essentials dna-advantage }] { subscription evaluation } [acceptEULA]</p> <p>Example:</p> <pre>Device(config)# license right-to-use activate ipbase acceptEULA Device(config)# license right-to-use activate addon dna-essentials subscription acceptEULA</pre>	<p>Activates the specified license level on a switch and enables acceptance of the end-user license agreement (EULA).</p> <p>To configure an add-on license, EULA acceptance is not mandatory.</p>
Step 4	<p>show license right-to-use usage</p> <p>Example:</p> <pre>Device(config)# show license right-to-use usage</pre>	<p>Displays detailed usage information.</p> <p>Other options are available with the show license right-to-use command.</p>

Rehosting a License

To rehost a license, you have to deactivate the license from one device and then activate the same license on another device. The following steps can be used to rehost a license.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **license right-to-use deactivate** [license-level] slotslot-num
4. **license right-to-use activate** [license-level]slot-num [**acceptEULA**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>license right-to-use deactivate [license-level] slotslot-num</p> <p>Example:</p> <pre>Device(config)# license right-to-use deactivate dna-essentials slot 1</pre>	<p>Deactivates the license on one device.</p>
Step 4	<p>license right-to-use activate [license-level]slot-num [acceptEULA]</p>	<p>Activates the license on another device.</p>

	Command or Action	Purpose
	Example: Device(config)# license right-to-use activate dna-essentials slot 2	

Monitoring Licenses

Use the following commands in the privilege EXEC mode to monitor license information:

Command	Purpose
show license right-to-use default	Displays the default license information.
show license right-to-use detail	Displays detailed information of all the licenses in the switch stack.
show license right-to-use eula	Displays the end user license agreement.
show license right-to-use slot slot-number	Displays the license information for a specific slot in a switch stack.
show license right-to-use summary	Displays a summary of the license information on the entire switch stack.
show license right-to-use usage [slot slot-number]	Displays detailed information about usage for all licenses in the switch stack.

Configuration Examples for License Levels

The following sections provide examples for configuring license levels.

Reference

.

Example: Displaying the detailed license information

The following examples shows how to display the detailed information of all the licenses in a stack using the **show license right-to-use detail** command.

```
Device# show license right-to-use detail
Index 1
  License Name      : Advanced Enterprise Services
  Period left      : Lifetime
  License Type     : permanent
  License State    : Active, In use
Index 2
  License Name      : dna-essentials
  Period left      : CSSM Managed
  License Type     : Subscription
  License State    : Not Activated
```

```

Index 3
License Name      : dna-advantage
Period left      : CSSM Managed
License Type     : Subscription
License State    : Active, In use

```

Example: Displaying a summary of the license information

The following examples shows how to display a summary of the license information using the **show license right-to-use summary** command.

```

Device# show license right-to-use summary
License Name      Type          Period left
-----
lanlite           Permanent  0 minute 0 second
lanbase           Permanent  0 minute 0 second
dna-essentials    Subscription CSSM Managed
-----

```

```

License Level In Use: lanbase  addon: dna-essentials
License Level on Reboot: lanbase  addon: dna-essentials

```

Example: show license right-to-use usage

```

FEX-0#show license right-to-use usage
slot      License Name      Type          In-use  EULA
-----
0          lanlite           Permanent     yes     yes
0          lanbase           Permanent     yes     yes
          dna-essentials    Subscription  yes     yes
          dna-advantage    Subscription  no      yes

```

Example: Displaying the end user license agreement

The following example shows how to display the end user license agreement.

```

Device# show license right-to-use eula subscription
Feature name      EULA Accepted
-----
dna-essentials    yes
dna-advantage     no
PLEASE READ THE FOLLOWING TERMS CAREFULLY. INSTALLING THE LICENSE OR
LICENSE KEY PROVIDED FOR ANY CISCO SOFTWARE PRODUCT, PRODUCT FEATURE,
AND OR SUBSEQUENTLY PROVIDED SOFTWARE FEATURES (COLLECTIVELY, THE ?SOFTWARE?),
USING SUCH SOFTWARE, AND/OR ACTIVATION OF THE SOFTWARE COMMAND LINE INTERFACE
CONSTITUTES YOUR FULL ACCEPTANCE OF THE FOLLOWING TERMS.YOU MUST NOT PROCEED
FURTHER IF YOU ARE NOT WILLING TO BE BOUND BY ALL THE TERMS SET FORTH HEREIN.

```

```

Your use of the Software is subject to the Cisco End User License Agreement (EULA)
and any relevant supplemental terms (SEULA) found at
http://www.cisco.com/c/en/us/about/legal/cloud-and-software/software-terms.html.
You hereby acknowledge and agree that certain Software and/or features are licensed
for a particular term, that the license to such Software and/or features is valid only
for the applicable term and that such Software and/or features may be shut down or
otherwise terminated by Cisco after expiration of the applicable license term (e.g.,
90-day trial period). Cisco reserves the right to terminate any such Software feature
electronically or by any other means available. While Cisco may provide alerts, it is
your sole responsibility to monitor your usage of any such term Software feature to
ensure that your systems and networks are prepared for a shutdown of the Software feature.

```

To memorialize your acceptance of these terms and activate your license to use the Software, please execute the command "license accept end user agreement force".

Feature History for Information About Licensing

Release	Modification
Cisco IOS Release 15.2(6)E1	This feature was introduced.



PART **XI**

Embedded Event Manager

- [Embedded Event Manager Overview, on page 1735](#)
- [Information About Writing EEM Policies Using the Cisco IOS CLI, on page 1757](#)
- [Writing Embedded Event Manager Policies Using Tcl, on page 1829](#)
- [Signed Tcl Scripts, on page 1889](#)
- [EEM CLI Library Command Extensions, on page 1913](#)
- [EEM Context Library Command Extensions, on page 1925](#)
- [EEM Event Registration Tcl Command Extensions, on page 1933](#)
- [EEM Event Tcl Command Extensions, on page 2017](#)
- [EEM Library Debug Command Extensions, on page 2025](#)
- [EEM Multiple Event Support Tcl Command Extensions, on page 2027](#)
- [EEM SMTP Library Command Extensions, on page 2031](#)
- [EEM System Information Tcl Command Extensions, on page 2035](#)
- [EEM Utility Tcl Command Extensions, on page 2047](#)



CHAPTER 87

Embedded Event Manager Overview

Embedded Event Manager (EEM) is a distributed and customized approach to event detection and recovery offered directly in a Cisco IOS device. EEM offers the ability to monitor events and take informational, corrective, or any desired EEM action when the monitored events occur or when a threshold is reached. An EEM policy is an entity that defines an event and the actions to be taken when that event occurs.

This module contains a technical overview of EEM. EEM can be used alone, or with other network management technologies to help monitor and maintain your network. Before you begin to implement EEM, it is important that you understand the information presented in this module.

- [Information About Embedded Event Manager, on page 1735](#)
- [Where to Go Next, on page 1753](#)
- [Feature Information for Embedded Event Manager 4.0 Overview, on page 1753](#)
- [Additional References, on page 1753](#)

Information About Embedded Event Manager

Embedded Event Manager

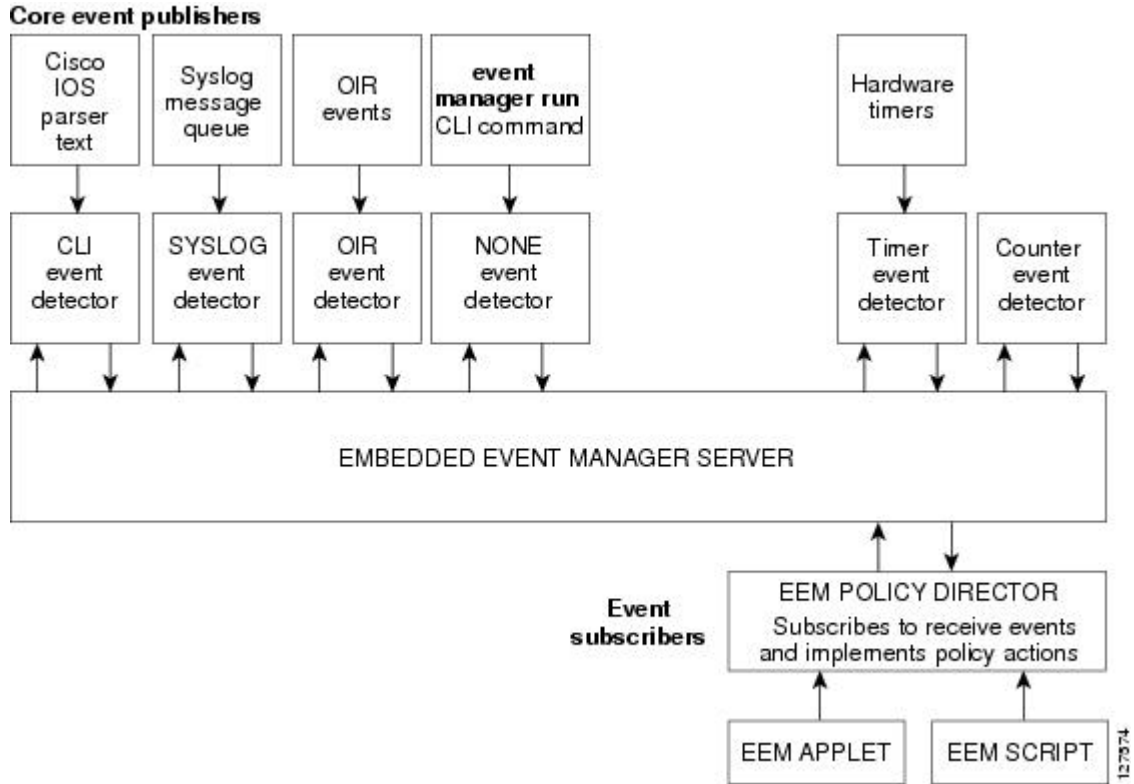
Event tracking and management has traditionally been performed by devices external to the networking device. Embedded Event Manager (EEM) has been designed to offer event management capability directly in Cisco IOS devices. The on-device, proactive event management capabilities of EEM are useful because not all event management can be done off device because some problems compromise communication between the device and the external network management device. Capturing the state of the device during such situations can be invaluable in taking immediate recovery actions and gathering information to perform root-cause analysis. Network availability is also improved if automatic recovery actions are performed without the need to fully reboot the routing device.

EEM is a flexible, policy-driven framework that supports in-box monitoring of different components of the system with the help of software agents known as event detectors. The figure below shows the relationship between the EEM server, core event publishers (event detectors), and the event subscribers (policies). Basically, event publishers screen events and publish them when there is a match on an event specification that is provided by the event subscriber. Event detectors notify the EEM server when an event of interest occurs. The EEM policies that are configured using the Cisco command-line interface (CLI) then implement recovery on the basis of the current state of the system and the actions specified in the policy for the given event.

EEM offers the ability to monitor events and take informational or corrective action when the monitored events occur or when a threshold is reached. An EEM policy is an entity that defines an event and the actions

to be taken when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the CLI configuration. A script is a form of policy that is written in Tool Command Language (Tcl).

Figure 131: Embedded Event Manager Core Event Detectors



Note If your network has a higher version of EEM, that version would include the previous releases of EEM version as well.

Embedded Event Manager 1.0

EEM 1.0 introduced the following event detectors:

- **SNMP**—The Simple Network Management Protocol (SNMP) event detector allows a standard SNMP MIB object to be monitored and an event to be generated when the object matches specified values or crosses specified thresholds.
- **Syslog**—The syslog event detector allows for screening syslog messages for a regular expression pattern match.

EEM 1.0 introduced the following actions:

- Generating prioritized syslog messages.
- Generating a Cisco Networking Services (CNS) event for upstream processing by CNS devices.

- Reloading the Cisco software.
- Switching to a secondary processor in a fully redundant hardware configuration.

Embedded Event Manager 2.0

EEM 2.0 introduced some new features. EEM 2.0 introduced the following event detectors:

- **Application-Specific**—The application-specific event detector allows any Embedded Event Manager policy to publish an event.
- **Counter**—The counter event detector publishes an event when a named counter crosses a specified threshold.
- **Interface Counter**—The interface counter event detector publishes an event when a generic Cisco IOS interface counter for a specified interface crosses a defined threshold.
- **Timer**—The timer event detector publishes events for the following four different types of timers: absolute-time-of-day, countdown, watchdog, and CRON.
- **Watchdog System Monitor (IOSWDSysMon)**—The Cisco IOS watchdog system monitor event detector publishes an event when CPU or memory utilization for a Cisco IOS process crosses a threshold.

EEM 2.0 introduced the following actions:

- Setting or modifying a named counter.
- Publishing an application-specific event
- Generating an SNMP trap.

The ability to run a Cisco defined sample policy written using Tool Command Language (Tcl) was introduced. A sample policy was provided that could be stored in the system policy directory.

Embedded Event Manager 2.1

EEM 2.1 introduced the following new event detectors:

- **CLI**—The CLI event detector screens command-line interface (CLI) commands for a regular expression match.
- **None**—The none event detector publishes an event when the Cisco IOS **event manager run** command executes an EEM policy.
- **OIR**—The online insertion and removal (OIR) event detector publishes an event when a particular hardware insertion or removal event occurs.

EEM 2.1 introduced the following actions:

- Executing a Cisco CLI command.
- Requesting system information when an event occurs.
- Sending a short e-mail.
- Manually running an EEM policy.

EEM 2.1 also permits multiple concurrent policies to be run using the new **event manager scheduler script** command. Support for SNMP event detector rate-based events is provided as is the ability to create policies using Tool Command Language (Tcl).

Embedded Event Manager 2.1 (Software Modularity)

EEM 2.1 (Software Modularity) is supported on Cisco Software Modularity images. EEM 2.1 (Software Modularity) introduced the following event detectors:

- **GOLD**—The Generic Online Diagnostic (GOLD) event detector publishes an event when a GOLD failure event is detected on a specified card and subcard.
- **System Manager**—The system manager event detector generates events for Cisco IOS Software Modularity process start, normal or abnormal stop, and restart events. The events generated by the system manager allows policies to change the default behavior of the process restart.
- **Watchdog System Monitor (WDSysMon)**—The Cisco Software Modularity watchdog system monitor event detector detects infinite loops, deadlocks, and memory leaks in Cisco IOS Software Modularity processes.

EEM 2.1 for Software Modularity introduced the ability to display EEM reliability metric data for processes.



Note EEM 2.1 for Software Modularity images also supports the resource and RF event detectors introduced in EEM 2.2, but it does not support the enhanced object tracking event detector or the actions to read and set tracked objects.

Embedded Event Manager 2.2

EEM 2.2 introduced some new features. EEM 2.2 introduced the following event detectors:

- **Enhanced Object Tracking**—The enhanced object tracking event detector publishes an event when the tracked object changes. Enhanced object tracking provides complete separation between the objects to be tracked and the action to be taken by a client when a tracked object changes.
- **Resource**—The resource event detector publishes an event when the Embedded Resource Manager (ERM) reports an event for the specified policy.
- **RF**—The redundancy framework (RF) event detector publishes an event when one or more RF events occur during synchronization in a dual Route Processor (RP) system. The RF event detector can also detect an event when a dual RP system continuously switches from one RP to another RP (referred to as a ping-pong situation).

EEM 2.2 introduced the following actions:

- Reading the state of a tracked object.
- Setting the state of a tracked object.

Embedded Event Manager 2.3

EEM 2.3 is supported on the Cisco Catalyst 6500 Series switches and introduces enhancements to the Generic Online Diagnostics (GOLD) Event Detector on that product.

- The **event gold** command was enhanced with the addition of the **action-notify**, **testing-type**, **test-name**, **test-id**, **consecutive-failure**, **platform-action**, and **maxrun** keywords for improved reaction to GOLD test failures and conditions.
- The following platform-wide GOLD Event Detector information can be accessed through new read-only EEM built-in environment variables:
 - Boot-up diagnostic level
 - Card index, name, serial number
 - Port counts
 - Test counts
- The following test-specific GOLD Event Detector information can be accessed through new read-only EEM built-in environment variables (available to EEM applets only):
 - Test name, attribute, total run count
 - Test result per test, port, or device
 - Total failure count, last fail time
 - Error code
 - Occurrence of consecutive failures

These enhancements result in reduced mean time to recovery (MTTR) and higher availability through improved automation and fault detection.

Embedded Event Manager 2.4

EEM 2.4 introduced the following event detectors:

- **SNMP Notification**—The SNMP notification event detector provides the ability to intercept SNMP trap and inform messages coming into the device. An SNMP notification event is generated when an incoming SNMP trap or inform message matches specified values or crosses specified thresholds.
- **RPC**—The remote procedure call (RPC) event detector provides the ability to invoke EEM policies from outside the device over an encrypted connection using Secure Shell (SSH). The RPC event detector uses Simple Object Access Protocol (SOAP) data encoding for exchanging XML-based messages. This event detector can be used to run EEM policies and then receive output in a SOAP XML-formatted reply.

EEM 2.4 added enhancements to the following event detectors:

- **Interface counter rate-based trigger**—This feature adds the ability for an interface event to be triggered based on a rate of change over a period of time. A rate can be specified both for the entry value and the exit value. This feature copies the rate-based functionality that currently exists for the SNMP event detector.
- **SNMP delta value**—The difference between the monitored Object Identifier (OID) value at the beginning of the monitored period and the actual OID value when the event is published will be provided in the **event reqinfo** data for both the SNMP event detector and the Interface Counter event detector.

EEM 2.4 introduced the following actions:

- Multiple event support—The ability to run multiple events was introduced, and **show event manager** commands were enhanced to show multiple events.
- Support for parameters—The *parameter* argument has been added to the **event manager run** command. A maximum of 15 parameters can be used.
- Display of Job IDs and completion status--Some of the **show event manager** commands were enhanced to display Job IDs and completion status.
- Bytecode support—Tcl 8 defines a specialized bytecode language (BCL) and includes a just-in-time compiler that translates Tcl scripts to BCL. Byte sequence is executed by a “virtual machine,” `Tcl_ExecuteByteCode()`, or TEBC for short, as often as needed. Currently EEM accepts file extensions, such as *.tcl for user policies and *.tm for system policies. Tcl standard extension for bytecode scripts are *.tbc. Now EEM will accept *.tbc as valid EEM policies.
- Registration substitution enhancement—Supports replacing multiple parameters in the event registration statement lines with a single environment variable.
- Tcl package support

Embedded Event Manager 3.0

EEM 3.0 introduced the following new event detectors:

- Custom CLI--The custom CLI event detector publishes an event to add and enhance existing CLI command syntax.
- Routing--The Routing event detector publishes an event when route entries change in the Routing Information Base (RIB).
- NetFlow-- The NetFlow event detector publishes an event when a NetFlow event is triggered.
- IP SLA--The IP SLA event detector publishes an event when an IP SLA reaction is triggered.

EEM 3.0 introduced the following features:

- Class-based scheduling--The EEM policies will be assigned a class using the **class** keyword when they are registered. EEM policies registered without a class will be assigned to the default class.
- High performance Tcl policies--Three new Tcl commands are introduced **event_completion**, **event_wait**, and **event_completion_with_wait**.
- Interactive cli support--The synchronous applets are enhanced to support interaction with the local console (TTY). Two new IOS commands, **action gets** and **action puts**, are introduced to allow users to enter and display input directly on the console.
- Variable logic for applets--The Variable Logic for EEM Applets feature adds the ability to apply conditional logic within EEM applets. Conditional logic introduces a control structure that can change the flow of actions within applets depending on conditional expressions.
- Digital signature support--A new API performs digital signature verification for a Tcl script to check if the script is signed by Cisco before execution.
- Support authenticating e-mail servers--The **action mail** command is modified to include an optional username and password.

- SMTP IPv6 support--The keyword **sourceaddr** is added in Tcl e-mail templates to specify either an IPv6 or IPv4 address.
- SNMP library extensions--The EEM applet **action info** and Tcl **sys_reqinfo_snmp** commands are enhanced to include functionality for SNMP getid, inform, trap, and set-type operations.
- SNMP Notification IPv6 support--IPv6 address is supported for the source and destination IP addresses.
- CLI Library XML-PI support--Provides a programmable interface which encapsulates IOS command-line interface (CLI) show commands in XML format in a consistent way across different Cisco products. Customers using XML-PI will be able to parse IOS show command output from within Tcl scripts using well-known keywords instead of having to depend on the use of regular expression support.

Embedded Event Manager 3.1

EEM 3.1 introduced one new event detector:

- SNMP Object--The Simple Network Management Protocol (SNMP) object trap event detector provides an extension to replace the value when an SNMP trap with the specified SNMP object ID (OID) is encountered on a specific interface or address.

EEM 3.1 added an enhancement to the following event detector:

- SNMP Notification--The SNMP notification event detector now can wait and intercept the outgoing SNMP traps and informs.

EEM 3.1 added enhancement to the following action:

- Specify facility--The **action syslog** command has been enhanced to specify syslog facility.

EEM 3.1 introduces the following features:

- Provides the ability to create a short description for the registered policy--A new **description** command has been introduced to register policies with a brief description in Cisco IOS CLI and Tcl policies. The **show event manager policy available** command and the **show event manager policy registered** command have been enhanced to add the **description** keyword to display the description of the registered applet.
- Enables EEM policies to bypass AAA authorization--The **event manager application** command has been enhanced to provide authorization and bypass keywords to disable AAA.
- Introduces CLI Library enhancements--Provides two new commands in the CLI library: **cli_run** and **cli_run_interactive**.

Embedded Event Manager 3.2

EEM 3.2 introduced the following new event detectors:

- Neighbor Discovery--Neighbor Discovery event detector provides the ability to publish a policy to respond to automatic neighbor detection when:
 - a Cisco Discovery Protocol (CDP) cache entry is added, deleted or updated.
 - a Link Layer Discovery Protocol (LLDP) cache entry is added, deleted, or updated.
 - an interface link status changes.

- an interface line status changes.
- Identity--Identity event detector generates an event when AAA authorization and authentication is successful, when failure occurs, or after normal user traffic on the port is allowed to flow.
- Mac-Address-Table--Mac-Address-Table event detector generates an event when a MAC address is learned in the MAC address table.



Note The Mac-Address-Table event detector is supported only on switch platforms and can be used only on Layer 2 interfaces where MAC addresses are learned. Layer 3 interfaces do not learn addresses and devices do not usually support the mac-address-table infrastructure needed to notify EEM of a learned MAC address.

EEM 3.2 also introduces new CLI commands to support the applets to work with the new event detectors.

Embedded Event Manager 4.0

EEM 4.0 introduces the following new features:

- EEM Email Action Enhancements
 - TLS support for SMTP mail actions—The new optional **secure** keyword is added to the **action mail** CLI with **tls** and **none** keyword options. There are no updates to the corresponding Tcl Policy.
 - Custom port for SMTP mail actions—The new optional **port** keyword is added to the **action mail** CLI. In the Tcl policy, the port number can be specified by adding a line to the e-mail template.
- EEM Security Enhancements
 - Checksum-based script integrity—Where digital signature is not supported or unavailable, users can still enforce some basic integrity check on the TCL policy by using the Unix command **openssl sha1**. The new optional **checksum**, **md5**, and **sha-1** keywords have been added to the **event manager policy** command.
 - Third-party digital signature support—Requires Tcl secure mode and a trustpoint to associate with the TCL scripts in order to verify the signature.
 - Script owner identification—If a policy is successfully registered with a digital signature, the owner (or signer) of the policy can be identified by using the **show event manager policy registered** command and checking the **Dsig** keyword in the show output.
 - Registration of remote Tcl policies—The new optional **remote** keyword has been added to the **event manager policy** command.
- EEM Resource Management
 - Resource consumption throttling—The new optional **resource-limit** keyword has been added to the **event manager scheduler** command.
 - Rate limiting of triggered policies per event—The new optional **rate-limit** keyword has been added to the **event syslog** command.
- EEM Usability Enhancements
 - File operations in EEM applet actions—The new CLI **action file** has been added to allow file selection.
 - New fields are added in EEM to track statistics of queue size, dropped events, and run-time using the **show event manager statistics** EXEC command. A set of new clear commands—**clear event**

manager detector counters and **clear event manager server counters** —are introduced to clear the event manager queue counters.

- EEM Event Detector Enhancements
 - CLI event detector enhancement—Provides the ability to detect the session where the user enters the event cli command. Four new keywords and built-in environmental variables—**username**, **host**, **privilege**, and **tty**— are added to the **event cli** applet and event_reqinfo array names to the **event_register_cli** event detector. The **show event manager detector EXEC** command has also been modified to reflect the enhancement.
 - Syslog event detector performance enhancement—Provides the option to perform string matching on specific log message fields. The four new keywords—**facility**, **mnemonic**, **sequence**, and **timestamp** keywords— are added to the **action syslog** command, **event syslog** command, and to the **event_register_syslog** event detector. The **show event manager detector EXEC** command has also been modified to reflect the enhancement.

EEM Event Detectors Available by Cisco IOS Release

EEM uses software programs known as event detectors to determine when an EEM event occurs. Some event detectors are available on every Cisco IOS release, but most event detectors have been introduced in a specific release. Use the table below to determine which event detectors are available in your specific Cisco IOS release. A blank entry (--) indicates that the event detector is not available; the text “Yes” indicates that the event detector is available. The event detectors shown in the table are supported in later releases of the same Cisco IOS release train. For more details on each event detector, see the Event Detectors concept in the “Embedded Event Manager Overview” module.

Table 175: Availability of Event Detectors by Cisco IOS Release

Event Detector	122(25)S	12.3(14)T 122(18)SXF5 12.2(28)SB 12.2(33)SRA	12.4(2)T 12.2(31)SB3 12.2(33)SRB	122(18)SXF4 Cisco IOS Software Modularity	122(33)SXH	12.4(20)T 122(33)SXI	12.4(22)T 122(33)SRE	15.0(1)M 15.1(3)T	15.2(5)S	15 E XE 3E
Application-Specific	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CLI	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	--	Yes
Counter	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Custom CLI	--	--	--	--	--	--	Yes	Yes	--	--
Enhanced Object Tracking	--	--	Yes	--	Yes	Yes	Yes	Yes	--	--
Environmental	--	--	--	--	--	--	--	--	--	Yes
GOLD	--	--	--	Yes	Yes	Yes	Yes	Yes	--	Yes
Identity	--	--	--	--	--	--	--	Yes	Yes	Yes
Interface Counter	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	--	Yes

Event Detector	12.2(25)S	12.3(14)T 12.2(18)SXF5 12.2(28)SB 12.2(33)SRA	12.4(2)T 12.2(31)SB3 12.2(33)SRB	12.2(18)SXF4 Cisco IOS Software Modularity	12.2(33)SXH	12.4(20)T 12.2(33)SXI	12.4(22)T 12.2(33)SRE	15.0(1)M 15.1(3)T	15.2(5)S 15.2(5)SY	15 E XE 3E
IPSLA	--	--	--	--	--	--	Yes	Yes	--	Yes
Mac-Address-Table	--	--	--	--	--	--	--	Yes	Yes	Yes
Neighbor Discovery	--	--	--	--	--	--	--	Yes	Yes	Yes
NF	--	--	--	--	--	--	Yes	Yes	--	--
None	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
OIR	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Resource	--	--	Yes	Yes	Yes	Yes	Yes	Yes	--	--
RF	--	--	Yes	Yes	Yes	Yes	Yes	Yes	--	Yes
Routing	--	--	--	--	--	--	Yes	Yes	--	Yes
RPC	--	--	--	--	--	Yes	Yes	Yes	Yes	--
SNMP	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	--	Yes
SNMP Proxy	--	--	--	--	--	--	--	--	Yes	--
SNMP Notification	--	--	--	--	--	Yes	Yes	Yes	--	Yes
SNMP Object	--	--	--	--	--	--	--	Yes	--	Yes
Syslog	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
System Manager	--	--	--	Yes	Yes	Yes	Yes	Yes	Yes	--
Timer	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
IOSWDSysMon (Cisco IOS watchdog)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	--	Yes
WDSysMon (Cisco IOS Software Modularity watchdog)	--	--	--	Yes	--	--	--	--	--	--

Event Detectors

Embedded Event Manager (EEM) uses software programs known as *event detectors* to determine when an EEM event occurs. Event detectors are separate systems that provide an interface between the agent being monitored, for example Simple Network Management Protocol (SNMP), and the EEM policies where an action can be implemented. Some event detectors are available on every Cisco IOS release, but most event detectors have been introduced in a specific release. For details of which event detector is supported in each Cisco IOS release, see the EEM Event Detectors Available by Cisco IOS Release concept in the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” or the “Writing Embedded Event Manager Policies Using Tcl” modules. EEM contains the following event detectors.

Application-Specific Event Detector

The application-specific event detector allows any Embedded Event Manager policy to publish an event. When an EEM policy publishes an event it must use an EEM subsystem number of 798 with any event type. If an existing policy is registered for subsystem 798 and a specified event type, a second policy of the same event type will trigger the first policy to run when the specified event is published.

CLI Event Detector

The CLI event detector screens command-line interface (CLI) commands for a regular expression match. When a match is found, an event is published. The match logic is performed on the fully expanded CLI command after the command is successfully parsed and before it is executed. The CLI event detector supports three publish modes:

- Synchronous publishing of CLI events--The CLI command is not executed until the EEM policy exits, and the EEM policy can control whether the command is executed. The read/write variable, `_exit_status`, allows you to set the exit status at policy exit for policies triggered from synchronous events. If `_exit_status` is 0, the command is skipped, if `_exit_status` is 1, the command is run.
- Asynchronous publishing of CLI events--The CLI event is published, and then the CLI command is executed.
- Asynchronous publishing of CLI events with command skipping--The CLI event is published, but the CLI command is not executed.

Counter Event Detector

The counter event detector publishes an event when a named counter crosses a specified threshold. There are two or more participants that affect counter processing. The counter event detector can modify the counter, and one or more subscribers define the criteria that cause the event to be published. After a counter event has been published, the counter monitoring logic can be reset to start monitoring the counter immediately or it can be reset when a second threshold--called an exit value--is crossed.

Custom CLI Event Detector

The custom CLI event detector publishes an event to add and enhance existing CLI command syntax. When the special parser characters Tab, ? (question mark), and Enter are entered, the parser sends the input to the custom CLI event detector for processing. The custom CLI event detector then compares this input against registered strings to determine if this is a new or enhanced CLI command. Upon a match the custom CLI event detector takes appropriate actions, such as displaying help for the command if ? is entered, displaying the entire command if Tab is entered, or executing the command if Enter was entered. If a match does not occur, the parser regains control and processes the information as usual.

Enhanced Object Tracking Event Detector

The enhanced object tracking (EOT) event detector publishes an event when the status of a tracked object changes. Object tracking was first introduced into the Hot Standby Router Protocol (HSRP) as a simple tracking mechanism that allowed you to track the interface line-protocol state only. If the line-protocol state of the interface went down, the HSRP priority of the device was reduced, allowing another HSRP device with a higher priority to become active.

Object tracking was enhanced to provide complete separation between the objects to be tracked and the action to be taken by a client when a tracked object changes. Thus, several clients such as HSRP, VRRP, or GLBP can register their interest with the tracking process, track the same object, and each take different action when the object changes. Each tracked object is identified by a unique number that is specified on the tracking command-line interface (CLI). Client processes use this number to track a specific object. The tracking process periodically polls the tracked objects and notes any change of value. The changes in the tracked object are communicated to interested client processes, either immediately or after a specified delay. The object values are reported as either up or down.

Enhanced object tracking is now integrated with EEM to allow EEM to report on a status change of a tracked object and to allow enhanced object tracking to track EEM objects. A new type of tracking object--a stub object--is created. The stub object can be manipulated using the existing CLI commands that already allow tracked objects to be manipulated.

GOLD Event Detector

The GOLD event detector publishes an event when a GOLD failure event is detected on a specified card and subcard.

Interface Counter Event Detector

The interface counter event detector publishes an event when a generic Cisco IOS interface counter for a specified interface crosses a defined threshold. A threshold can be specified as an absolute value or an incremental value. If the incremental value is set to 50, for example, an event would be published when the interface counter increases by 50.

After an interface counter event has been published, the interface counter monitoring logic is reset using two methods. The interface counter is reset either when a second threshold--called an exit value--is crossed or when an elapsed period of time occurs.

IP SLA Event Detector

The IP SLA event detector publishes an event when an IP SLA reaction is triggered.

NetFlow Event Detector

The NetFlow event detector publishes an event when a NetFlow event is triggered.

None Event Detector

The none event detector publishes an event when the Cisco IOS **event manager run** CLI command executes an EEM policy. EEM schedules and runs policies on the basis of an event specification that is contained within the policy itself. An EEM policy must be identified and registered to be permitted to run manually before the **event manager run** command will execute.

OIR Event Detector

The online insertion and removal (OIR) event detector publishes an event when one of the following hardware insertion or removal event occurs:

- A card is removed.
- A card is inserted.

Route Processors (RPs), line cards, or feature cards can be monitored for OIR events.

Resource Event Detector

The resource event detector publishes an event when the Embedded Resource Manager (ERM) reports an event for the specified policy. The ERM infrastructure tracks resource depletion and resource dependencies across processes and within a system to handle various error conditions. The error conditions are handled by providing an equitable sharing of resources between various applications. The ERM framework provides a communication mechanism for resource entities and allows communication between these resource entities from numerous locations. The ERM framework also helps in debugging CPU and memory-related issues. The ERM monitors system resource usage to better understand scalability needs by allowing you to configure threshold values for resources such as the CPU, buffers, and memory. The ERM event detector is the preferred method for monitoring resources in Cisco software but the ERM event detector is not supported in Software Modularity images. For more details about ERM, go to “Embedded Resource Manager” module.

RF Event Detector

The redundancy framework (RF) event detector publishes an event when one or more RF events occur during synchronization in a dual Route Processor (RP) system. The RF event detector can also detect an event when a dual RP system continuously switches from one RP to another RP (referred to as a ping-pong situation).

RPC Event Detector

The remote procedure call (RPC) event detector provides the ability to invoke EEM policies from outside the device over an encrypted connection using Secure Shell (SSH). The RPC event detector uses Simple Object Access Protocol (SOAP) data encoding for exchanging XML-based messages. This event detector can be used to run EEM policies and then receive output in a SOAP XML-formatted reply.

Routing Event Detector

The routing event detector publishes an event when a route entry changes in the Routing Information Base (RIB).

SNMP Event Detector

The SNMP event detector allows a standard SNMP MIB object to be monitored and an event to be generated when the object matches specified values or crosses specified thresholds.

SNMP Notification Event Detector

The SNMP notification event detector provides the ability to intercept SNMP trap and inform messages coming into or going out of the device. An SNMP notification event is generated when an incoming or outgoing SNMP trap or inform message matches specified values or crosses specified thresholds. The SNMP event detector can wait and intercept the outgoing SNMP traps and informs.

SNMP Object Event Detector

The Simple Network Management Protocol (SNMP) object trap event detector provides an extension to replace the value when an SNMP trap with the specified SNMP object ID (OID) is encountered on a specific interface or address.

Syslog Event Detector

The syslog event detector allows for screening syslog messages for a regular expression pattern match. The selected messages can be further qualified, requiring that a specific number of occurrences be logged within a specified time. A match on a specified event criteria triggers a configured policy action.

System Manager Event Detector

The system manager event detector generates events for Cisco IOS Software Modularity process start, normal or abnormal stop, and restart events. The events generated by the system manager allows policies to change the default behavior of the process restart.

Timer Event Detector

The timer event detector publishes events for the following four different types of timers:

- An absolute-time-of-day timer publishes an event when a specified absolute date and time occurs.
- A countdown timer publishes an event when a timer counts down to zero.
- A watchdog timer publishes an event when a timer counts down to zero and then the timer automatically resets itself to its initial value and starts to count down again.
- A CRON timer publishes an event using a UNIX standard CRON specification to indicate when the event is to be published. A CRON timer never publishes events more than once per minute.

Watchdog System Monitor (IOSWDSysMon) Event Detector for Cisco IOS

The Cisco IOS watchdog system monitor event detector publishes an event when one of the following occurs:

- CPU utilization for a Cisco IOS task crosses a threshold.
- Memory utilization for a Cisco IOS task crosses a threshold.



Note Cisco IOS processes are now referred to as tasks to distinguish them from Cisco IOS Software Modularity processes.

Two events may be monitored at the same time, and the event publishing criteria can be specified to require one event or both events to cross their specified thresholds.

Watchdog System Monitor (WDSysMon) Event Detector for Cisco IOS Software Modularity

The Cisco IOS Software Modularity watchdog system monitor event detector detects infinite loops, deadlocks, and memory leaks in Cisco IOS Software Modularity processes.

EEM Actions Available by Cisco IOS Release

The CLI-based corrective actions that are taken when event detectors report events enable a powerful on-device event management mechanism. Some actions are available in every Cisco IOS release, but most actions have been introduced in a specific release. Use the table below to determine which actions are available in your specific Cisco IOS release. A blank entry (--) indicates that the action is not available; the text “Yes” indicates that the action is available. The actions shown in the table are supported in later releases of the same Cisco IOS release train. For more details on each action, see the Embedded Event Manager Actions concept in the “Embedded Event Manager Overview” module.

Table 176: Availability of Actions by Cisco IOS Release

Action	12.2(25)S	12.3(14)T 12.2(18)SXF5 12.2(28)SB 12.2(33)SRA	12.4(2)T 12.2(31)SB3 12.2(33)SRB	12.2(18)SXF4 Cisco IOS Software Modularity	12.2(33)SXH	12.4(20)T	12.4(22)T	15.0(1)M	15E XE 3E
Execute a CLI command	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Generate a CNS event	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Generate a prioritized syslog message	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Generate an SNMP trap	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Manually run an EEM policy	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Publish an application-specific event	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Read the state of a tracked object	--	--	Yes	--		Yes	Yes	Yes	Yes
Reload the Cisco software	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Request system information	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Send a short e-mail	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Set or modify a named counter	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Set the state of a tracked object	--	--	Yes	--		Yes	Yes	Yes	Yes
Switch to a secondary RP	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Embedded Event Manager Actions

The CLI-based corrective actions that are taken when event detectors report events enable a powerful on-device event management mechanism. Some EEM actions are available on every Cisco IOS release, but most EEM actions have been introduced in a specific release. For details of which EEM action is supported in each Cisco IOS release, see the EEM Actions Available by Cisco IOS Release concept in the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” or the “Writing Embedded Event Manager Policies Using Tcl” modules. EEM supports the following actions:

- Executing a Cisco IOS command-line interface (CLI) command.
- Generating a CNS event for upstream processing by Cisco CNS devices.
- Setting or modifying a named counter.
- Switching to a secondary processor in a fully redundant hardware configuration.
- Requesting system information when an event occurs.
- Sending a short e-mail.
- Manually running an EEM policy.
- Publishing an application-specific event.
- Reloading the Cisco software.
- Generating an SNMP trap.
- Generating prioritized syslog messages.
- Reading the state of a tracked object.
- Setting the state of a tracked object.

EEM action CLI commands contain an EEM action label that is a unique identifier that can be any string value. Actions are sorted and run in ascending alphanumeric (lexicographical) key sequence using the label as the sort key. If you are using numbers as labels be aware that alphabetical sorting will sort 10.0 after 1.0, but before 2.0, and in this situation we recommend that you use numbers such as 01.0, 02.0, and so on, or use an initial letter followed by numbers.

Embedded Event Manager Environment Variables

EEM allows environment variables to be used in EEM policies. Tool Command Language (Tcl) allows global variables to be defined that are known to all procedures within a Tcl script. EEM allows environment variables to be defined using a CLI command, the **event manager environment** command, for use within an EEM policy. All EEM environment variables are automatically assigned to Tcl global variables before a Tcl script is run. There are three different types of environment variables associated with Embedded Event Manager:

- User-defined--Defined by you if you create an environment variable in a policy that you have written.
- Cisco-defined--Defined by Cisco for a specific sample policy.
- Cisco built-in (available in EEM applets)--Defined by Cisco and can be read only or read/write. The read only variables are set by the system before an applet starts to execute. The single read/write variable, `_exit_status`, allows you to set the exit status at policy exit for policies triggered from synchronous events.

Cisco-defined environment variables (see the table below) and Cisco system-defined environment variables may apply to one specific event detector or to all event detectors. Environment variables that are user-defined or defined by Cisco in a sample policy are set using the **event manager environment** command. Variables that are used in the EEM policy must be defined before you register the policy. A Tcl policy contains a section called “Environment Must Define” that can be defined to check that any required environment variables are defined before the policy runs.

Cisco built-in environment variables are a subset of the Cisco-defined environment variables and the built-in variables are available to EEM applets only. The built-in variables can be read-only or can be read and write, and these variables may apply to one specific event detector or to all event detectors. For more details and a table listing the Cisco system-defined variables, see the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.



Note Cisco-defined environment variables begin with an underscore character (_). We strongly recommend that customers avoid the same naming convention to prevent naming conflicts.

The table below describes the Cisco-defined variables used in the sample EEM policies. Some of the environment variables do not have to be specified for the corresponding sample policy to run and these are marked as optional.

Table 177: Cisco-Defined Environmental Variables and Examples

Environment Variable	Description	Example
_config_cmd1	The first configuration command that is executed.	interface Ethernet1/0
_config_cmd2	(Optional) The second configuration command that is executed.	no shutdown
_crash_reporter_debug	(Optional) A value that identifies whether debug information for tm_crash_reporter.tcl will be enabled.	1
_crash_reporter_url	The URL location to which the crash report is sent.	http://www.yourdomain.com/fm/interface_tm.cgi
_cron_entry	A CRON specification that determines when the policy will run. See the “Writing Embedded Event Manager Policies Using Tcl” module for more information about how to specify a cron entry.	0-59/1 0-23/1 * * 0-7
_email_server	A Simple Mail Transfer Protocol (SMTP) mail server used to send e-mail.	mailserver.yourdomain.com
_email_to	The address to which e-mail is sent.	engineer@yourdomain.com
_email_from	The address from which e-mail is sent.	devtest@yourdomain.com
_email_cc	The address to which the e-mail is be copied.	manager@yourdomain.com

Environment Variable	Description	Example
_email_ipaddr	The source IP address of the recipient.	209.165.201.1 or (IPv6 address) 2001:0DB8::1
_info_snmp_oid	The SNMP object ID.	1.3.6.1.2.1.2 or iso.internet.mgmt.mib-2.interfaces
_info_snmp_value	The value string of the associated SNMP data element.	
_show_cmd	The CLI show command to be executed when the policy is run.	show version
_syslog_pattern	A regular expression pattern match string that is used to compare syslog messages to determine when the policy runs.	.*UPDOWN.*FastEthernet 0/0.*
_tm_fsys_usage_cron	(Optional) A CRON specification that is used in the event_register keyword extension. If unspecified, the <code>_tm_fsys_usage.tcl</code> policy is triggered once per minute.	0-59/1 0-23/1 * * 0-7
_tm_fsys_usage_debug	(Optional) When this variable is set to a value of 1, disk usage information is displayed for all entries in the system.	1
_tm_fsys_usage_freebytes	(Optional) Free byte threshold for systems or specific prefixes. If free space falls below a given value, a warning is displayed.	disk2:98000000
_tm_fsys_usage_percent	(Optional) Disk usage percentage thresholds for systems or specific prefixes. If disk usage percentage exceeds a given percentage, a warning is displayed. If unspecified, the default disk usage percentage is 80 percent for all systems.	nvrnram:25 disk2:5

Embedded Event Manager Policy Creation

EEM is a policy driven process in which the EEM policy engine receives notifications when faults and other events occur in the Cisco software system. Embedded Event Manager policies implement recovery based on the current state of the system and the actions specified in the policy for a given event. Recovery actions are triggered when the policy is run.

Although there are some EEM CLI configuration and **show** commands, EEM is implemented through the creation of policies. An EEM policy is an entity that defines an event and the actions to be taken when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the CLI configuration. A script is a form of policy that is written in Tcl.

The creation of an EEM policy involves:

- Selecting the event for which the policy is run.

- Defining the event detector options associated with logging and responding to the event.
- Defining the environment variables, if required.
- Choosing the actions to be performed when the event occurs.

There are two ways to create an EEM policy. The first method is to write applets using CLI commands, and the second method is to write Tcl scripts. Cisco provides enhancements to Tcl in the form of Tcl command extensions that facilitate the development of EEM policies. Scripts are defined off the networking device using an ASCII editor. The script is then copied to the networking device and registered with EEM. When a policy is registered with the Embedded Event Manager, the software examines the policy and registers it to be run when the specified event occurs. Policies can be unregistered or suspended. Both types of policies can be used to implement EEM in your network.

For details on writing EEM policies using the Cisco IOS CLI, go to “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.

For details on writing EEM policies using Tcl, go to “Writing Embedded Event Manager Policies Using Tcl” module.

Where to Go Next

- If you want to write EEM policies using the Cisco IOS CLI, see the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.
- If you want to write EEM policies using Tcl, see the “Writing Embedded Event Manager Policies Using Tcl” module.

Feature Information for Embedded Event Manager 4.0 Overview

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 178: Feature Information for Embedded Event Manager 4.0 Overview

Feature Name	Releases	Feature Information
Embedded Event Manager 4.0	IOS 15.2(5)E1	This feature was introduced and is supported only on c2960cx.

Additional References

The following sections provide references related to EEM.

Related Documents

Related Topic	Document Title
EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples	Cisco IOS Embedded Event Manager Command Reference
Embedded Event Manager policy writing using the CLI	Writing Embedded Event Manager Policies Using the Cisco IOS CLI module
Embedded Event Manager policy writing using Tcl	Writing Embedded Event Manager Policies Using Tcl module
Embedded Resource Manager	Embedded Resource Manager module

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	--

MIBs

MIB	MIBs Link
CISCO-EMBEDDED-EVENT-MGR-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	--

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>



CHAPTER 88

Information About Writing EEM Policies Using the Cisco IOS CLI

- [Prerequisites for Writing EEM Policies Using the Cisco IOS CLI, on page 1757](#)
- [Information About Writing EEM Policies Using the Cisco IOS CLI, on page 1757](#)
- [How to Write EEM Policies Using the Cisco IOS CLI, on page 1769](#)
- [Configuration Examples for Writing Embedded Event Manager Policies Using Tcl, on page 1812](#)
- [Additional References, on page 1827](#)
- [Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI, on page 1828](#)

Prerequisites for Writing EEM Policies Using the Cisco IOS CLI

- Before writing EEM policies, you should be familiar with the concepts explained in the “Embedded Event Manager Overview” module.
- If the **action cns-event** command is used, access to a Cisco Networking Services (CNS) Event gateway must be configured.
- If the **action force-switchover** command is used, a secondary processor must be configured on the device.
- If the **action snmp-trap** command is used, the **snmp-server enable traps event-manager** command must be enabled to permit SNMP traps to be sent from the Cisco IOS device to the SNMP server. Other relevant **snmp-server** commands must also be configured; for details see the **action snmp-trap** command page.

Information About Writing EEM Policies Using the Cisco IOS CLI

Embedded Event Manager Policies

EEM offers the ability to monitor events and take informational or corrective action when the monitored events occur or a threshold is reached. An EEM policy is an entity that defines an event and the actions to be

taken when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the CLI configuration. A script is a form of policy that is written in Tool Command Language (Tcl).

EEM Applet

An EEM applet is a concise method for defining event screening criteria and the actions to be taken when that event occurs. In applet configuration mode, three types of configuration statements are supported. The **event** commands are used to specify the event criteria to trigger the applet to run, the **action** commands are used to specify an action to perform when the EEM applet is triggered, and the **set** command is used to set the value of an EEM applet variable. Currently only the `_exit_status` variable is supported for the **set** command.

Only one **event** configuration command is allowed within an applet configuration. When applet configuration mode is exited and no **event** command is present, a warning is displayed stating that no event is associated with this applet. If no event is specified, this applet is not considered registered. When no action is associated with this applet, events are still triggered but no actions are performed. Multiple **action** configuration commands are allowed within an applet configuration. Use the **show event manager policy registered** command to display a list of registered applets.

Before modifying an EEM applet, be aware that the existing applet is not replaced until you exit applet configuration mode. While you are in applet configuration mode modifying the applet, the existing applet may be executing. It is safe to modify the applet without unregistering it. When you exit applet configuration mode, the old applet is unregistered and the new version is registered.

The action configuration commands are uniquely identified using the *label* argument, which can be any string value. Actions are sorted in ascending alphanumeric key sequence using the *label* argument as the sort key, and they are run using this sequence.

The Embedded Event Manager schedules and runs policies on the basis of an event specification that is contained within the policy itself. When applet configuration mode is exited, EEM examines the **event** and **action** commands that are entered and registers the applet to be run when a specified event occurs.

EEM Script

Scripts are defined off the networking device using an ASCII editor. The script is then copied to the networking device and registered with EEM. Tcl scripts are supported by EEM.

EEM allows you to write and implement your own policies using Tcl. Writing an EEM policy involves:

- Selecting the event for which the policy is run.
- Defining the event detector options associated with logging and responding to the event.
- Choosing the actions to be followed when the event occurs.

Cisco provides enhancements to Tcl in the form of keyword extensions that facilitate the development of EEM policies. The main categories of keywords identify the detected event, the subsequent action, utility information, counter values, and system information. For more details about writing EEM policies using Tcl, see the “Writing Embedded Event Manager Policies Using Tcl” module.

Embedded Event Manager Built-In Environment Variables Used in EEM Applets

EEM built-in environment variables are a subset of the Cisco-defined environment variables and the built-in variables are available to EEM applets only. The built-in variables can be read-only or can be read and write

and these variables may apply to one specific event detector or to all event detectors. The table below lists the Cisco built-in environment variables that are read-only alphabetically by event detector and subevent.

Table 179: EEM Built-In Environment Variables (Read Only)

Environment Variable	Description
All Events	
_event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
_event_type	
_event_type_string	Type of event.
_event_pub_sec _event_pub_msec	An ASCII string identifier of the event type that triggered the event.
_event_severity	The time, in seconds and milliseconds, at which the event was published to the EEM.
Application-Specific Event Detector	The severity of the event.
_application_component_id	The event application component identifier.
_application_data1	The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published.
_application_data2	The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published.
_application_data3	The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published.
_application_data4	The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published.
_application_sub_system	The event application subsystem number.
_application_type	The type of application.
CLI Event Detector	
_cli_msg	The fully expanded message that triggered the CLI event.
_cli_msg_count	The number of times that a message match occurred before the event was published.
Counter Event Detector	

Environment Variable	Description
_counter_name	The name of the counter.
_counter_value	The value of the counter.
Enhanced Object Tracking Event Detector	
_track_number	The number of the tracked object.
_track_state	The state of the tracked object; down or up.
GOLD Event Detector	
_action_notify	The action notify information in a GOLD event flag; either false or true.
_event_severity	The event severity which can be one of the following; normal, minor, or major.
_gold_bl	The boot diagnostic level, which can be one of the following values: <ul style="list-style-type: none"> • 0: complete diagnostic • 1: minimal diagnostic • 2: bypass diagnostic
_gold_card	The card on which a GOLD failure event was detected.
_gold_cf <i>testnum</i>	Consecutive failure, where <i>testnum</i> is the test number. For example, _gold_cf3 is the EEM built-in environment variable for consecutive failure of test 3.
_gold_ci	Card index.
_gold_cn	Card name.
_gold_ec <i>testnum</i>	Test error code, where <i>testnum</i> is the test number. For example, _gold_ec3 is the EEM built-in environment variable for the error code of test 3.
_gold_lf <i>testnum</i>	Last fail time, where <i>testnum</i> is the test number. For example, _gold_lf3 is the EEM built-in variable for the last fail time of test 3. The time-stamp format is <i>mmm dd yyyy hh:mm:ss</i> . For example, Mar 11 2005 08:47:00.
_gold_new_failure	The new test failure information in a GOLD event flag; either true or false.

Environment Variable	Description
_gold_overall_result	The overall diagnostic result, which can be one of the following values: <ul style="list-style-type: none"> • 0: OK • 3: minor error • 4: major error • 14: unknown result
_gold_pc	Port counts.
_gold_rc <i>testnum</i>	Test total run count, where <i>testnum</i> is the test number. For example, _gold_rc3 is the EEM built-in variable for the total run count of test 3.
_gold_sn	Card serial number.
_gold_sub_card	The subcard on which a GOLD failure event was detected.
_gold_ta <i>testnum</i>	Test attribute, where <i>testnum</i> is the test number. For example, _gold_ta3 is the EEM built-in variable for the test attribute of test 3.
_gold_tc	Test counts.
_gold_tf <i>testnum</i>	Total failure count, where <i>testnum</i> is the test number. For example, _gold_tf3 is the EEM built-in variable for the total failure count of test 3.
_gold_tn <i>testnum</i>	Test name, where <i>testnum</i> is the test number. For example, _gold_tn3 is the EEM built-in variable for the name of test 3.
_gold_tr <i>testnum</i>	Test result, where <i>testnum</i> is the test number. For example, _gold_tr6 is the EEM built-in variable for test 6, where test 6 is not a per-port test and not a per-device test. The test result is one of the following values: <ul style="list-style-type: none"> • P: diagnostic result Pass • F: diagnostic result Fail • U: diagnostic result Unknown

Environment Variable	Description
_gold_tr <i>testnum d devnum</i>	Per-device test result, where <i>testnum</i> is the test number and <i>devnum</i> is the device number. For example, _gold_tr3d20 is the EEM built-in variable for the test result for test 3, device 20. The test result is one of the following values: <ul style="list-style-type: none"> • P: diagnostic result Pass • F: diagnostic result Fail • U: diagnostic result Unknown
_gold_tr <i>testnum p portnum</i>	Per-port test result, where <i>testnum</i> is the test number and <i>portnum</i> is the port number. For example, _gold_tr5p20 is the EEM built-in variable for the test result for test 5, port 20. The test result is one of the following values: <ul style="list-style-type: none"> • P: diagnostic result Pass • F: diagnostic result Fail • U: diagnostic result Unknown
_gold_tt	The testing type, which can be one of the following: <ul style="list-style-type: none"> • 1: a boot diagnostic • 2: an on-demand diagnostic • 3: a schedule diagnostic • 4: a monitoring diagnostic
Interface Counter Event Detector	
_interface_is_increment	A value to indicate whether the current interface counter value is an absolute value (0) or an increment value (1).
_interface_name	The name of the interface to be monitored.
_interface_parameter	The name of the interface counter to be monitored.
_interface_value	A value with which the current interface counter value is compared.
None Event Detector	
_event_id	A value of 1 indicates an insertion event; a value of 2 indicates a removal event.

Environment Variable	Description
_none_arg _none_arg1 _none_arg2 _none_arg3 _none_arg4 _none_arg5 _none_arg6 _none_arg7 _none_arg8 _none_arg9 _none_arg10 _none_arg11 _none_arg12 _none_arg13 _none_arg14 _none_arg15	The parameters that are passed from the XML SOAP command to the script.
OIR Event Detector	
_oir_event	A value of 1 indicates an insertion event; a value of 2 indicates a removal event.
_oir_slot	The slot number for the OIR event.
Resource Event Detector	
_resource_configured_threshold	The configured ERM threshold.
_resource_current_value	The current value reported by ERM.
_resource_dampen_time	The ERM dampen time, in nanoseconds.
_resource_direction	The ERM event direction. The event direction can be one of the following: up, down, or no change.
_resource_level	The ERM event level. The four event levels are normal, minor, major, and critical.
_resource_notify_data_flag	The ERM notify data flag.
_resource_owner_id	The ERM resource owner ID.
_resource_policy_id	The ERM policy ID.

Environment Variable	Description
<code>_resource_policy_violation_flag</code>	The ERM policy violation flag; either false or true.
<code>_resource_time_sent</code>	The ERM event time, in nanoseconds.
<code>_resource_user_id</code>	The ERM resource user ID.
RF Event Detector	
<code>_rf_event</code>	A value of 0 indicates that this is not an RF event; a value of 1 indicates an RF event.
RPC Event Detector	
<code>_rpc_event</code>	A value of 0 indicates that there is no error; a value of 1 to 83 indicates error.
<code>_rpc_argc</code> <code>_rpc_arg0</code> <code>_rpc_arg1</code> <code>_rpc_arg2</code> <code>_rpc_arg3</code> <code>_rpc_arg4</code> <code>_rpc_arg5</code> <code>_rpc_arg6</code> <code>_rpc_arg7</code> <code>_rpc_arg8</code> <code>_rpc_arg9</code> <code>_rpc_arg10</code> <code>_rpc_arg11</code> <code>_rpc_arg12</code> <code>_rpc_arg13</code> <code>_rpc_arg14</code>	The parameters that are passed from the XML SOAP command to the applet.
SNMP Event Detector	
<code>_snmp_exit_event</code>	A value of 0 indicates that this is not an exit event; a value of 1 indicates an exit event.
<code>_snmp_oid</code>	The SNMP object ID that caused the event to be published.
<code>_snmp_oid_delta_val</code>	The actual incremental difference between the value of the current SNMP object ID and the value when the event was last triggered.

Environment Variable	Description
<code>_snmp_oid_val</code>	The SNMP object ID value when the event was published.
SNMP Notification Event Detector	
<code>_snmp_notif_oid</code>	A user specified object ID.
<code>_snmp_notif_oid_val</code>	A user specified object ID value.
<code>_snmp_notif_src_ip_addr</code>	The source IP address of the SNMP Protocol Data Unit (PDU).
<code>_snmp_notif_dest_ip_addr</code>	The destination IP address of the SNMP PDU.
<code>_x_x_x_x_x_x_x(varbinds)</code>	The SNMP PDU varbind information.
<code>_snmp_notif_trunc_vb_buf</code>	Indicates whether the varbind information has been truncated due to the lack of space in the buffer.
Syslog Event Detector	
<code>_syslog_msg</code>	The syslog message that caused the event to be published.
System Manager (Process) Event Detector	
<code>_process_dump_count</code>	The number of times that a Posix process was dumped.
<code>_process_exit_status</code>	The status of the Posix process at exit.
<code>_process_fail_count</code>	The number of times that a Posix process failed.
<code>_process_instance</code>	The instance number of the Posix process.
<code>_process_last_respawn</code>	The Posix process that was last respawned.
<code>_process_node_name</code>	The node name of the Posix process.
<code>_process_path</code>	The path of the Posix process.
<code>_process_process_name</code>	The name of the Posix process.
<code>_process_respawn_count</code>	The number of times that a Posix process was respawned.
Timer Event Detector	
<code>_timer_remain</code>	The time available before the timer expires. Note This environment variable is not available for the CRON timer.
<code>_timer_time</code>	The time at which the last event was triggered.
<code>_timer_type</code>	The type of timer.

Environment Variable	Description
Watchdog System Monitor (IOSWDSysMon) Event Detector	
_ioswd_node	The slot number for the Route Processor (RP) reporting node.
_ioswd_num_subs	The number of subevents present.
All Watchdog System Monitor (IOSWDSysMon) Subevents	
_ioswd_sub1_present _ioswd_sub2_present	A value to indicate whether subevent 1 or subevent 2 is present. A value of 1 means that the subevent is present; a value of 0 means that the subevent is not present.
_ioswd_sub1_type _ioswd_sub2_type	The event type, either <code>cpu_proc</code> or <code>mem_proc</code> .
Watchdog System Monitor (IOSWDSysMon) <code>cpu_proc</code> Subevents	
_ioswd_sub1_path _ioswd_sub2_path	A process name of subevents.
_ioswd_sub1_period _ioswd_sub2_period	The time period, in seconds and optional milliseconds, used for measurement in subevents.
_ioswd_sub1_pid _ioswd_sub2_pid	The process identifier of subevents.
_ioswd_sub1_taskname _ioswd_sub2_taskname	The task name of subevents.
_ioswd_sub1_value _ioswd_sub2_value	The CPU utilization of subevents measured as a percentage.
Watchdog System Monitor (IOSWDSysMon) <code>mem_proc</code> Subevents	
_ioswd_sub1_diff _ioswd_sub2_diff	A percentage value of the difference that triggered the event. Note This variable is set only when the _ioswd_sub1_is_percent or _ioswd_sub2_is_percent variable contains a value of 1.
_ioswd_sub1_is_percent _ioswd_sub2_is_percent	A number that identifies whether the value is a percentage. A value of 0 means that the value is not a percentage; a value of 1 means that the value is a percentage.
_ioswd_sub1_path _ioswd_sub2_path	The process name of subevents.
_ioswd_sub1_pid _ioswd_sub2_pid	The process identifier of subevents.
_ioswd_sub1_taskname _ioswd_sub2_taskname	The task name of subevents.
_ioswd_sub1_value _ioswd_sub2_value	The CPU utilization of subevents measured as a percentage.

Environment Variable	Description
Watchdog System Monitor (WDSysMon) Event Detector	
_wd_sub1_present _wd_sub2_present	A value to indicate whether subevent 1 or subevent 2 is present. A value of 1 means that the subevent is present; a value of 0 means that the subevent is not present.
_wd_num_subs	The number of subevents present.
_wd_sub1_type _wd_sub2_type	The event type: cpu_proc, cpu_tot, deadlock, dispatch_mgr, mem_proc, mem_tot_avail, or mem_tot_used.
Watchdog System Monitor (WDSysMon) cpu_proc Subevents	
_wd_sub1_node _wd_sub2_node	The slot number for the subevent RP reporting node.
_wd_sub1_period _wd_sub2_period	The time period, in seconds and optional milliseconds, used for measurement in subevents.
_wd_sub1_procname _wd_sub2_procname	The process name of subevents.
_wd_sub1_value _wd_sub2_value	The CPU utilization of subevents measured as a percentage.
Watchdog System Monitor (WDSysMon) cpu_tot Subevents	
_wd_sub1_node _wd_sub2_node	The slot number for the subevent RP reporting node.
_wd_sub1_period _wd_sub2_period	The time period, in seconds and optional milliseconds, used for measurement in subevents.
_wd_sub1_value _wd_sub2_value	The CPU utilization of subevents measured as a percentage.
Watchdog System Monitor (WDSysMon) deadlock Subevents	
_wd_sub1_entry_[1-N]_b_node _wd_sub2_entry_[1-N]_b_node	The slot number for the subevent RP reporting node.
_wd_sub1_entry_[1-N]_b_pid _wd_sub2_entry_[1-N]_b_pid	The process identifier of subevents.
_wd_sub1_entry_[1-N]_b_procname _wd_sub2_entry_[1-N]_b_procname	The process name of subevents.
_wd_sub1_entry_[1-N]_b_tid _wd_sub2_entry_[1-N]_b_tid	The time identifier of subevents.
_wd_sub1_entry_[1-N]_node _wd_sub2_entry_[1-N]_node	The slot number for the subevent RP reporting node.

Environment Variable	Description
<code>_wd_sub1_entry_[1-N]_pid</code> <code>_wd_sub2_entry_[1-N]_pid</code>	The process identifier of subevents.
<code>_wd_sub1_entry_[1-N]_procname</code> <code>_wd_sub2_entry_[1-N]_procname</code>	The process name of subevents.
<code>_wd_sub1_entry_[1-N]_state</code> <code>_wd_sub2_entry_[1-N]_state</code>	The time identifier of subevents.
<code>_wd_sub1_entry_[1-N]_tid</code> <code>_wd_sub2_entry_[1-N]_tid</code>	The time identifier of subevents.
<code>_wd_sub1_num_entries</code> <code>_wd_sub2_num_entries</code>	The number of subevents.
Watchdog System Monitor (WDSysMon) dispatch manager Subevents	
<code>_wd_sub1_node</code> <code>_wd_sub2_node</code>	The slot number for the subevent RP reporting node.
<code>_wd_sub1_period</code> <code>_wd_sub2_period</code>	The time period, in seconds and optional milliseconds, used for measurement in subevents.
<code>_wd_sub1_procname</code> <code>_wd_sub2_procname</code>	The process name of subevents.
<code>_wd_sub1_value</code> <code>_wd_sub2_value</code>	The CPU utilization of subevents measured as a percentage.
Watchdog System Monitor (WDSysMon) mem_proc Subevents	
<code>_wd_sub1_diff</code> <code>_wd_sub2_diff</code>	A percentage value of the difference that triggered the event. Note This variable is set only when the <code>_wd_sub1_is_percent</code> or <code>_wd_sub2_is_percent</code> variable contains a value of 1.
<code>_wd_sub1_is_percent</code> <code>_wd_sub2_is_percent</code>	A number that identifies whether the value is a percentage. A value of 0 means that the value is not a percentage; a value of 1 means that the value is a percentage.
<code>_wd_sub1_node</code> <code>_wd_sub2_node</code>	The slot number for the subevent RP reporting node.
<code>_wd_sub1_period</code> <code>_wd_sub2_period</code>	The time period, in seconds and optional milliseconds, used for measurement in subevents.
<code>_wd_sub1_pid</code> <code>_wd_sub2_pid</code>	The process identifier of subevents.
<code>_wd_sub1_procname</code> <code>_wd_sub2_procname</code>	The process name of subevents.
<code>_wd_sub1_value</code> <code>_wd_sub2_value</code>	The CPU utilization of subevents measured as a percentage.

Environment Variable	Description
Watchdog System Monitor (WDSysMon) mem_tot_avail and mem_tot_used Subevents	
_wd_sub1_avail _wd_sub2_avail	The memory available for subevents.
_wd_sub1_diff _wd_sub2_diff	A percentage value of the difference that triggered the event. Note This variable is set only when the <code>_wd_sub1_is_percent</code> or <code>_wd_sub2_is_percent</code> variable contains a value of 1.
_wd_sub1_is_percent _wd_sub2_is_percent	A number that identifies whether the value is a percentage. A value of 0 means that the value is not a percentage; a value of 1 means that the value is a percentage.
_wd_sub1_node _wd_sub2_node	The slot number for the subevent RP reporting node.
_wd_sub1_period _wd_sub2_period	The time period, in seconds and optional milliseconds, used for measurement in subevents.
_wd_sub1_value _wd_sub2_value	The CPU utilization of subevents measured as a percentage.
_wd_sub1_used _wd_sub2_used	The memory used by subevents.

How to Write EEM Policies Using the Cisco IOS CLI

Registering and Defining an Embedded Event Manager Applet

Perform this task to register an applet with Embedded Event Manager and to define the EEM applet using the Cisco IOS CLI **event** and **action** commands. Only one **event** command is allowed in an EEM applet. Multiple **action** commands are permitted. If no **event** and no **action** commands are specified, the applet is removed when you exit configuration mode.

The SNMP event detector and the syslog **action** commands used in this task are just representing any event detector and **action** commands. For examples using other event detectors and **action** commands, see the [Embedded Event Manager Applet Configuration Examples, on page 1812](#).

EEM Environment Variables

EEM environment variables for EEM policies are defined using the EEM **event manager environment** configuration command. By convention, all Cisco EEM environment variables begin with “_”. In order to avoid future conflict, customers are urged not to define new variables that start with “_”.

You can display the EEM environment variables set on your system by using the **show event manager environment** privileged EXEC command.

For example, you can create EEM policies that can send e-mails when an event occurs. The table below describes the e-mail-specific environment variables that can be used in EEM policies.

Table 180: EEM E-mail-Specific Environmental Variables

Environment Variable	Description
<code>_email_server</code>	A Simple Mail Transfer Protocol (SMTP) mail server used to send e-mail.
<code>_email_to</code>	The address to which e-mail is sent.
<code>_email_from</code>	The address from which e-mail is sent.
<code>_email_cc</code>	The address to which the e-mail is copied.

Alphabetical Order of EEM Action Labels

An EEM action label is a unique identifier that can be any string value. Actions are sorted and run in ascending alphanumeric (lexicographical) key sequence using the label as the sort key. If you are using numbers as labels be aware that alphanumerical sorting will sort 10.0 after 1.0, but before 2.0, and in this situation we recommend that you use numbers such as 01.0, 02.0, and so on, or use an initial letter followed by numbers.

SUMMARY STEPS

1. **enable**
2. **show event manager environment** [**all**] *variable-name*
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. Repeat [Alphabetical Order of EEM Action Labels](#) for all the required environment variables.
6. **event manager applet** *applet-name*
7. Do one of the following:
 - **event snmp oid** *oid-value* **get-type** {**exact**|**next**} **entry-op** *operator* **entry-val** *entry-value*[**exit-comb**|**and**]} [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
8. **action** *label* **cli command** *cli-string* [**pattern** *pattern-string*]
9. **action** *label* **syslog** [**priority** *priority-level*] **msg** *msg-text* **facility** *string*
10. **action** *label* **mail server** *server-address* **to** *to-address* **from** *from-address* [**cc** *cc-address*] **subject** *subject* **body** *body-text*
11. Add more action commands as required.
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device> enable</pre>	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>show event manager environment [all] <i>variable-name</i>]</p> <p>Example:</p> <pre>Device# show event manager environment all</pre>	<p>(Optional) Displays the name and value of EEM environment variables.</p> <ul style="list-style-type: none"> • The optional all keyword displays all the EEM environment variables. • The optional <i>variable-name</i> argument displays information about the specified environment variable.
Step 3	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 4	<p>event manager environment <i>variable-name string</i></p> <p>Example:</p> <pre>Device(config)# event manager environment _email_to engineering@example.com</pre>	<p>Configures the value of the specified EEM environment variable.</p> <ul style="list-style-type: none"> • In this example, the environment variable that holds the e-mail address to which e-mail is sent is set to <code>engineering@example.com</code>.
Step 5	<p>Repeat Alphabetical Order of EEM Action Labels for all the required environment variables.</p>	<p>Repeat Alphabetical Order of EEM Action Labels to configure all the environment variables required by the policy to be registered in Alphabetical Order of EEM Action Labels.</p>
Step 6	<p>event manager applet <i>applet-name</i></p> <p>Example:</p> <pre>Device(config)# event manager applet memory-fail</pre>	<p>Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.</p>
Step 7	<p>Do one of the following:</p> <ul style="list-style-type: none"> • event snmp oid <i>oid-value</i> get-type {exact next} entry-op <i>operator</i> entry-val <i>entry-value</i> [exit-comb and] [exit-op <i>operator</i>] [exit-val <i>exit-value</i>] [exit-time <i>exit-time-value</i>] poll-interval <i>poll-int-value</i> <p>Example:</p> <pre>Device(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val 512000 poll-interval 90</pre>	<p>Specifies the event criteria that cause the EEM applet to run.</p> <ul style="list-style-type: none"> • In this example, an EEM event is triggered when free memory falls below the value of 5120000. • Exit criteria are optional, and if not specified, event monitoring is reenabled immediately.
Step 8	<p>action <i>label</i> cli command <i>cli-string</i> [pattern <i>pattern-string</i>]</p>	<p>Specifies the action of executing a Cisco IOS CLI command when an EEM applet is triggered.</p>

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config-applet)# action 1.0 cli command "enable"</pre> <p>Example:</p> <pre>Device(config-applet)# action 2.0 cli command "clear counters Ethernet0/1" pattern "confirm"</pre> <p>Example:</p> <pre>Device(config-applet)# action 3.0 cli command "y"</pre>	<p>The pattern keyword is optional and is used only when the command string solicits input. The action cli command ends when the solicited prompt as specified in the optional pattern keyword is received. You are required to specify a regular expression pattern that will match the next solicited prompt. Specification of an incorrect pattern will cause the action cli command to wait forever until the applet execution times out due to the maxrun timer expiration.</p> <ul style="list-style-type: none"> The action taken is to specify an EEM applet to run when the pattern keyword specifies the <i>confirm</i> argument for the clear counters Ethernet0/1 command. In this case the command string solicits input, such as “confirm,” which has to be completed with a “yes” or a “no” input.
Step 9	<p>action label syslog [priority priority-level] msg msg-text facility string</p> <p>Example:</p> <pre>Device(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current available memory is \$snmp_oid_val bytes"</pre> <p>Example:</p> <pre>Device(config-applet)# action 1.0 syslog priority errors facility EEM-FAC message "TEST MSG"</pre>	<p>Specifies the action to be taken when an EEM applet is triggered.</p> <p>In this example, the action taken is to write a message to syslog.</p> <ul style="list-style-type: none"> The optional priority keyword specifies the priority level of the syslog messages. If selected, the <i>priority-level</i> argument must be defined. The <i>msg-text</i> argument can be character text, an environment variable, or a combination of the two. The facility keyword specifies the location of generated message The <i>string</i> argument can be character text, an environment variable, or a combination of the two.
Step 10	<p>action label mail server server-address to to-address from from-address [cc cc-address] subject subject body body-text</p> <p>Example:</p> <pre>Device(config-applet)# action 2.0 mail server 192.168.1.10 to engineering@example.com from devtest@example.com subject "Memory failure" body "Memory exhausted; current available memory is \$snmp_oid_val bytes"</pre>	<p>Specifies the action of sending a short e-mail when an EEM applet is triggered.</p> <ul style="list-style-type: none"> The <i>server-address</i> argument specifies the fully qualified domain name of the e-mail server to be used to forward the e-mail. The <i>to-address</i> argument specifies the e-mail address where the e-mail is to be sent. The <i>from-address</i> argument specifies the e-mail address from which the e-mail is sent. The <i>subject</i> argument specifies the subject line content of the e-mail as an alphanumeric string. The <i>body-text</i> argument specifies the text content of the e-mail as an alphanumeric string.

	Command or Action	Purpose
Step 11	Add more action commands as required.	--
Step 12	end Example: <pre>Device(config-applet)# end</pre>	Exits applet configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

Use the **debug event manager** command in privileged EXEC mode to troubleshoot EEM command operations. Use any debugging command with caution as the volume of generated output can slow or stop the device operations. We recommend that this command be used only under the supervision of a Cisco engineer.

Registering and Defining an EEM Tcl Script

Perform this task to configure environment variables and register an EEM policy. EEM schedules and runs policies on the basis of an event specification that is contained within the policy itself. When an EEM policy is registered, the software examines the policy and registers it to be run when the specified event occurs.

Before you begin

You must have a policy available that is written in the Tcl scripting language. Sample policies are provided--see the details in the [Sample EEM Policies, on page 1847](#) to see which policies are available for the Cisco IOS release image that you are using--and these sample policies are stored in the system policy directory.

SUMMARY STEPS

1. **enable**
2. **show event manager environment** [**all**| *variable-name*]
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. Repeat [Registering and Defining an EEM Tcl Script](#) to configure all the environment variables required by the policy to be registered in [Registering and Defining an EEM Tcl Script](#).
6. **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager environment [all <i>variable-name</i>] Example:	(Optional) Displays the name and value of EEM environment variables.

	Command or Action	Purpose
	Device# show event manager environment all	<ul style="list-style-type: none"> The optional all keyword displays all the EEM environment variables. The optional <i>variable-name</i> argument displays information about the specified environment variable.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	event manager environment <i>variable-name string</i> Example: Device(config)# event manager environment _cron_entry 0-59/2 0-23/1 * * 0-6	Configures the value of the specified EEM environment variable. <ul style="list-style-type: none"> In this example, the software assigns a CRON timer environment variable to be set to the second minute of every hour of every day.
Step 5	Repeat Registering and Defining an EEM Tcl Script to configure all the environment variables required by the policy to be registered in Registering and Defining an EEM Tcl Script .	--
Step 6	event manager policy <i>policy-filename</i> [type { system user }] [trap] Example: Device(config)# event manager policy tm_cli_cmd.tcl type system	Registers the EEM policy to be run when the specified event defined within the policy occurs. <ul style="list-style-type: none"> Use the system keyword to register a Cisco-defined system policy. Use the user keyword to register a user-defined system policy. Use the trap keyword to generate an SNMP trap when the policy is triggered. In this example, the sample EEM policy named <code>tm_cli_cmd.tcl</code> is registered as a system policy.
Step 7	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Examples

In the following example, the **show event manager environment** privileged EXEC command is used to display the name and value of all EEM environment variables.

```
Device# show event manager environment all
No.  Name                               Value
```

```

1  _cron_entry          0-59/2 0-23/1 * * 0-6
2  _show_cmd           show ver
3  _syslog_pattern     .*UPDOWN.*Ethernet1/0.*
4  _config_cmd1       interface Ethernet1/0
5  _config_cmd2       no shut

```

Unregistering Embedded Event Manager Policies

Perform this task to remove an EEM policy from the running configuration file. Execution of the policy is canceled.

SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [description *[policy-name]* | **detailed** *policy-filename* [system | user] | [event-type *event-name*] [system | user] [time-ordered | name-ordered]]
3. **configure terminal**
4. **no event manager policy** *policy-filename*
5. **exit**
6. Repeat Step 2 to ensure that the policy has been removed.

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager policy registered [description <i>[policy-name]</i> detailed <i>policy-filename</i> [system user] [event-type <i>event-name</i>] [system user] [time-ordered name-ordered]] Example: Device# show event manager policy registered	(Optional) Displays the EEM policies that are currently registered. <ul style="list-style-type: none"> • The optional system and user keywords display the registered system and user policies. • If no keywords are specified, EEM registered policies for all event types are displayed in time order.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	no event manager policy <i>policy-filename</i> Example: Device(config)# no event manager policy IPSLAping1	Removes the EEM policy from the configuration, causing the policy to be unregistered.

	Command or Action	Purpose
Step 5	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 6	Repeat Step 2 to ensure that the policy has been removed. Example: Device# show event manager policy registered	--

Examples

In the following example, the **show event manager policy registered** privileged EXEC command is used to display the two EEM applets that are currently registered:

```
Device# show event manager policy registered
No.  Class  Type      Event Type      Trap  Time Registered      Name
1    applet  system   snmp            Off   Fri Aug 12 17:42:52 2005  IPSLAping1
oid {1.3.6.1.4.1.9.9.42.1.2.9.1.6.4} get-type exact entry-op eq entry-val {1}
exit-op eq exit-val {2} poll-interval 90.000
action 1.0 syslog priority critical msg "Server IPEcho Failed: OID=$_snmp_oid_val"
action 1.1 snmp-trap strdata "EEM detected server reachability failure to 10.1.88.9"
action 1.2 publish-event sub-system 88000101 type 1 arg1 "10.1.88.9" arg2 "IPSLAEcho"
arg3 "fail"
action 1.3 counter name _IPSLA1F op inc value 1
2    applet  system   snmp            Off   Thu Sep 15 05:57:16 2005  memory-fail
oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
action 1.0 syslog priority critical msg Memory exhausted; current available memory is
$_snmp_oid_val bytes
action 2.0 force-switchover
```

In the following example, the **show event manager policy registered** privileged EXEC command is used to show that applet IPSLAping1 has been removed after entering the **no event manager policy** command:

```
Device# show event manager policy registered
No.  Class  Type      Event Type      Trap  Time Registered      Name
1    applet  system   snmp            Off   Thu Sep 15 05:57:16 2005  memory-fail
oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
action 1.0 syslog priority critical msg Memory exhausted; current available memory is
$_snmp_oid_val bytes
action 2.0 force-switchover
```

Suspending All Embedded Event Manager Policy Execution

Perform this task to immediately suspend the execution of all EEM policies. Suspending policies, instead of unregistering them might be necessary for reasons of temporary performance or security.

SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [description [*policy-name*] | **detailed** *policy-filename* [system | user] | [event-type *event-name*] [system | user] [time-ordered | name-ordered]]
3. **configure terminal**
4. **event manager scheduler suspend**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager policy registered [description [<i>policy-name</i>] detailed <i>policy-filename</i> [system user] [event-type <i>event-name</i>] [system user] [time-ordered name-ordered]] Example: Device# show event manager policy registered	(Optional) Displays the EEM policies that are currently registered. <ul style="list-style-type: none"> • The optional system and user keywords display the registered system and user policies. • If no keywords are specified, EEM registered policies for all event types are displayed in time order.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	event manager scheduler suspend Example: Device(config)# event manager scheduler suspend	Immediately suspends the execution of all EEM policies.
Step 5	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Displaying Embedded Event Manager History Data

Perform this optional task to change the size of the history tables and to display EEM history data.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **event manager history size** {events | traps} [size]
4. **exit**
5. **show event manager history events** [detailed] [maximum number]
6. **show event manager history traps** {server | policy}

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 configure terminal

Enters global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 event manager history size {events | traps} [size]

Use this command to change the size of the EEM event history table or the size of the EEM SNMP trap history table. In the following example, the size of the EEM event history table is changed to 30 entries:

Example:

```
Device(config)# event manager history size events 30
```

Step 4 exit

Exits global configuration mode and returns to privileged EXEC mode.

Example:

```
Device(config)# exit
```

Step 5 show event manager history events [detailed] [maximum number]

Use this command to display detailed information about each EEM event, for example:

Example:

```
Device# show event manager history events
No.  Time of Event      Event Type      Name
1    Fri Aug13  21:42:57 2004  snmp            applet: SAAping1
2    Fri Aug13  22:20:29 2004  snmp            applet: SAAping1
3    Wed Aug18  21:54:48 2004  snmp            applet: SAAping1
4    Wed Aug18  22:06:38 2004  snmp            applet: SAAping1
5    Wed Aug18  22:30:58 2004  snmp            applet: SAAping1
6    Wed Aug18  22:34:58 2004  snmp            applet: SAAping1
7    Wed Aug18  22:51:18 2004  snmp            applet: SAAping1
8    Wed Aug18  22:51:18 2004  application     applet: CustApp1
```

Step 6 show event manager history traps {server | policy}

Use this command to display the EEM SNMP traps that have been sent either from the EEM server or from an EEM policy. In the following example, the EEM SNMP traps that were triggered from within an EEM policy are displayed.

Example:

```
Device# show event manager history traps policy
No.   Time           Trap Type      Name
1     Wed Aug18  22:30:58 2004  policy        EEM Policy Director
2     Wed Aug18  22:34:58 2004  policy        EEM Policy Director
3     Wed Aug18  22:51:18 2004  policy        EEM Policy Director
```

Displaying Embedded Event Manager Registered Policies

Perform this optional task to display registered EEM policies.

SUMMARY STEPS

1. **enable**
2. **show event manager policy registered [event-type *event-name*] [time-ordered| name-ordered]**

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 **show event manager policy registered [event-type *event-name*] [time-ordered| name-ordered]**

Use this command with the **time-ordered** keyword to display information about currently registered policies sorted by time, for example:

Example:

```
Device# show event manager policy registered time-ordered
No.  Type      Event Type      Time           Registered Name
1    applet   snmp             Thu May30 05:57:16 2004 memory-fail
    oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val
    {5120000} poll-interval 90
    action 1.0 syslog priority critical msg "Memory exhausted; current available memory
    is $_snmp_oid_val bytes"
    action 2.0 force-switchover
2    applet   syslog          Wed Jul16 00:05:17 2004 intf-down
    pattern {.*UPDOWN.*Ethernet1/0.*}
    action 1.0 cns-event msg "Interface state change: $_syslog_msg"
```

Use this command with the **name-ordered** keyword to display information about currently registered policies sorted by name, for example:

Example:

```

Device# show event manager policy registered name-ordered
No.  Type   Event Type           Time Registered      Name
1    applet  syslog                Wed Jul16  00:05:17 2004  intf-down
    pattern {.*UPDOWN.*Ethernet1/0.*}
    action 1.0 cns-event msg "Interface state change: $_syslog_msg"
2    applet  snmp                  Thu May30  05:57:16 2004  memory-fail
    oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val
    {5120000} poll-interval 90
    action 1.0 syslog priority critical msg "Memory exhausted; current available memory
is $_snmp_oid_val bytes"
    action 2.0 force-switchover

```

Use this command with the **event-type** keyword to display information about currently registered policies for the event type specified in the *event-name* argument, for example:

Example:

```

Device# show event manager policy registered event-type syslog
No.  Type   Event Type           Time Registered      Name
1    applet  syslog                Wed Jul16  00:05:17 2004  intf-down
    pattern {.*UPDOWN.*Ethernet1/0.*}
    action 1.0 cns-event msg "Interface state change: $_syslog_msg"

```

Configuring Event SNMP Notification

Perform this task to configure SNMP notifications.

Before you begin

- SNMP event manager must be configured using the **snmp-server manager** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event** [**tag** *event-tag*] **snmp-notification oid** *oid-string* **oid-val** *comparison-value* **op** *operator* [**maxrun** *maxruntime-number*] [**src-ip-address** *ip-address*] [**dest-ip-address** *ip-address*] [**default** *seconds*] [**direction** {**incoming** | **outgoing**}] [**msg-op** {**drop** | **send**}]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet snmp	Registers the applet with the event manager server and enters applet configuration mode.
Step 4	event [tag <i>event-tag</i>] snmp-notification oid <i>oid-string</i> oid-val <i>comparison-value</i> op <i>operator</i> [maxrun <i>maxruntime-number</i>] [src-ip-address <i>ip-address</i>] [dest-ip-address <i>ip-address</i>] [default <i>seconds</i>] [direction { incoming outgoing }] [msg-op { drop send }] Example: Device(config-applet)# event snmp-notification dest-ip-address 192.168.1.1 oid 1 op eq oid-val 10	Specifies the event criteria for an Embedded Event Manager (EEM) applet that is run by sampling Simple Network Management Protocol (SNMP) notification.
Step 5	end Example: Device(config-applet)# end	Exits applet configuration mode and returns to privileged EXEC mode.

Configuring Multiple Event Support

The multiple event support feature adds the ability to register multiple events in the EEM server. The multiple event support involves one or more event occurrences, one or more tracked object states, and a time period for the event to occur. The event parameters are specified in the CLI commands. The data structure to handle multiple events contains multiple event identifiers and correlation logic. This data is used to register multiple events in the EEM Server.

Setting the Event Configuration Parameters

The **trigger** command enters the trigger applet configuration mode and specifies the multiple event configuration statements for EEM applets. The trigger statement is used to relate multiple event statement using the *tag* argument specified in each event statement. The events are raised based on the specified parameters.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event** [**tag** *event-tag*] **cli pattern** *regular-expression* **sync** {**yes** | **no skip** {**yes** | **no**}} [**occurs** *num-occurrences*] [**period** *period-value*] [**maxrun** *maxruntime-number*]
5. **trigger** [**occurs** *occurs-value*] [**period** *period-value*] [**period-start** *period-start-value*] [**delay** *delay-value*]

6. **correlate** {**event** *event-tag* | **track** *object-number*} [*boolean-operator* **event** *event-tag*]
7. **attribute** **tag** *event-tag* [**occurs** *occurs-value*]
8. **action** *label* **cli command** *cli-string*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet EventInterface	Registers an applet with EEM and enters applet configuration mode.
Step 4	event [tag <i>event-tag</i>] cli pattern <i>regular-expression</i> sync { yes no skip { yes no }} [occurs <i>num-occurrences</i>] [period <i>period-value</i>] [maxrun <i>maxruntime-number</i>] Example: Device(config-applet)# event tag 1.0 cli pattern "show bgp all" sync yes occurs 32 period 60 maxrun 60	Specifies the event criteria for an EEM applet that is run by matching a Cisco IOS command-line interface (CLI) command.
Step 5	trigger [occurs <i>occurs-value</i>] [period <i>period-value</i>] [period-start <i>period-start-value</i>] [delay <i>delay-value</i>] Example: Device(config-applet)# trigger occurs 1 period-start "0 8 * * 1-5" period 60	Specifies the complex event configuration parameters for an EEM applet.
Step 6	correlate { event <i>event-tag</i> track <i>object-number</i> } [<i>boolean-operator</i> event <i>event-tag</i>] Example: Device(config-applet)# correlate event 1.0 or event 2.0	Specifies a complex event correlation in the trigger mode for an EEM applet. Note When "and" is used to group events such as traps or syslog messages, then the default trigger occurrence window is three minutes.
Step 7	attribute tag <i>event-tag</i> [occurs <i>occurs-value</i>] Example: Device(config-applet)# attribute tag 1.0 occurs 1	Specifies up to eight attribute statements to build a complex event for an EEM applet.

	Command or Action	Purpose
Step 8	action <i>label</i> cli command <i>cli-string</i> Example: Device(config-applet)# action 1.0 cli command "show pattern"	Specifies the action of executing a CLI command when an EEM applet is triggered.

Examples

In the following example, applet is run if the **show bgp all** CLI command and any syslog message that contains the string "COUNT" occurred within a period 60 seconds.

```

event manager applet delay_50
  event tag 1.0 cli pattern "show bgp all" sync yes occurs 32 period 60 maxrun 60
  event tag 2.0 syslog pattern "COUNT"
  trigger occurs 1 delay 50
  correlate event 1.0 or event 2.0
  attribute tag 1.0 occurs 1
  attribute tag 2.0 occurs 1
  action 1.0 cli command "show pattern"
  action 2.0 cli command "enable"
  action 3.0 cli command "config terminal"
  action 4.0 cli command " ip route 192.0.2.0 255.255.255.224 192.0.2.12"
  action 91.0 cli command "exit"
  action 99.0 cli command "show ip route | incl 192.0.2.5"

```

Configuring EEM Class-Based Scheduling

To schedule Embedded Event Manager (EEM) policies and set policy scheduling options, perform this task. In this task, two EEM execution threads are created to run applets assigned to the default class.

The EEM policies will be assigned a class using the **class** keyword when they are registered. EEM policies registered without a class will be assigned to the default class. Threads that have default class, will service the default class when the thread is available for work. Threads that are assigned specific class letters will service any policy with a matching class letter when the thread is available for work.

If there is no EEM execution thread available to run the policy in the specified class and a scheduler rule for the class is configured, the policy will wait until a thread of that class is available for execution. Synchronous policies that are triggered from the same input event should be scheduled in the same execution thread.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager scheduler** {*applet* | *axp* | *call-home*} **thread class** *class-options* **number** *thread-number*
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager scheduler { applet axp call-home } thread class <i>class-options</i> number <i>thread-number</i> Example: Device(config)# event manager scheduler applet thread class default number 2	Schedules EEM policies and sets policy scheduling options. <ul style="list-style-type: none"> • In this example, two EEM execution threads are created to run applets assigned to the default class.
Step 4	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Holding a Scheduled EEM Policy Event or Event Queue

To hold a scheduled EEM policy event or event queue in the EEM scheduler, perform this task. In this task, all pending EEM policies are displayed. A policy identified using a job ID of 2 is held in the EEM scheduler, and the final step shows that the policy with a job ID of 2 has changed status from pending to held.

SUMMARY STEPS

1. **enable**
2. **show event manager policy pending** [queue-type {**applet** | **call-home** | **axp** | **script**} class *class-options* | **detailed**]
3. **event manager scheduler hold** {**all** | **policy** *job-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**} class *class-options*} [**processor** {**rp_primary** | **rp_standby**}]
4. **show event manager policy pending** [queue-type {**applet** | **call-home** | **axp** | **script**} class *class-options* | **detailed**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>show event manager policy pending [queue-type {applet call-home axp script} class class-options detailed]</p> <p>Example:</p> <pre>Device# show event manager policy pending</pre>	Displays the pending EEM policies.
Step 3	<p>event manager scheduler hold {all policy job-id queue-type {applet call-home axp script} class class-options} [processor {rp_primary rp_standby}]</p> <p>Example:</p> <pre>Device# event manager scheduler hold policy 2</pre>	<p>Holds a scheduled EEM policy event or event queue in the EEM scheduler.</p> <ul style="list-style-type: none"> In this example, a policy with a job ID of 2 is put on hold.
Step 4	<p>show event manager policy pending [queue-type {applet call-home axp script} class class-options detailed]</p> <p>Example:</p> <pre>Device# show event manager policy pending</pre>	Displays the status of EEM policy put on hold in Step 3 as held, along with other pending policies.

Examples

The following example shows how to view all pending EEM policies and to hold the EEM policy with a job ID of 2.

```
Device# show event manager policy pending
no. job id status time of event          event type    name
1  1      pend  Thu Sep 7 02:54:04 2006  syslog      applet: one
2  2      pend  Thu Sep 7 02:54:04 2006  syslog      applet: two
3  3      pend  Thu Sep 7 02:54:04 2006  syslog      applet: three
Device# event manager scheduler hold policy 2
Device# show event manager policy pending

no. job id status time of event          event type    name
1  1      pend  Thu Sep 7 02:54:04 2006  syslog      applet: one
2  2      held  Thu Sep 7 02:54:04 2006  syslog      applet: two
3  3      pend  Thu Sep 7 02:54:04 2006  syslog      applet: three
```

Resuming Execution of EEM Policy Events or Event Queues

To resume the execution of specified EEM policies, perform this task. In this task, the policy that was put on hold in the Holding a Scheduled EEM Policy Event or Event Queue task is now allowed to resume execution.

SUMMARY STEPS

- enable
- show event manager policy pending
- event manager scheduler release {all | policy policy-id | queue-type {applet | call-home | axp | script}} class class-options [processor {rp_primary | rp_standby}]
- show event manager policy pending

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager policy pending Example: Device# show event manager policy pending	Displays the pending and held EEM policies. Note Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.
Step 3	event manager scheduler release {all policy <i>policy-id</i> queue-type {applet call-home axp script}} class <i>class-options</i> [processor {rp_primary rp_standby}] Example: Device# event manager scheduler release policy 2	Resumes execution of specified EEM policies. <ul style="list-style-type: none"> • The example shows how to resume the execution of the policy with job ID of 2.
Step 4	show event manager policy pending Example: Device# show event manager policy pending	Displays the status of the EEM policy resumed in Step 3 as pending, along with other pending policies. Note Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.

Examples

The following example shows how to view all pending EEM policies, to specify the policy that will resume execution, and to see that the policy is now back in a pending status.

```

Device# show event manager policy pending

no. job id status time of event          event type      name
1 1      pend  Thu Sep 7 02:54:04 2006  syslog         applet: one
2 2      held   Thu Sep 7 02:54:04 2006  syslog         applet: two
3 3      pend   Thu Sep 7 02:54:04 2006  syslog         applet: three
Rotuer# event manager scheduler release policy 2
Rotuer# show event manager policy pending

no. job id status time of event          event type      name
1 1      pend  Thu Sep 7 02:54:04 2006  syslog         applet: one
2 2      pend  Thu Sep 7 02:54:04 2006  syslog         applet: two
3 3      pend  Thu Sep 7 02:54:04 2006  syslog         applet: three

```

Clearing Pending EEM Policy Events or Event Queues

Perform this task to clear EEM policies that are executing or pending execution. In this task, the EEM policy with a job ID of 2 is cleared from the pending queue. The **show event manager policy pending** command is used to display the policies that are pending before and after the policy is cleared.

SUMMARY STEPS

1. **enable**
2. **show event manager policy pending**
3. **event manager scheduler clear** {all | policy *job-id* | queue-type {applet | call-home | axp | script} class *class-options*} [processor {rp_primary | rp_standby}]
4. **show event manager policy pending**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager policy pending Example: Device# show event manager policy pending	Displays the pending EEM policies. Note Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.
Step 3	event manager scheduler clear {all policy <i>job-id</i> queue-type {applet call-home axp script} class <i>class-options</i> } [processor {rp_primary rp_standby}] Example: Device# event manager scheduler clear policy 2	Clears EEM policies that are executing or pending execution. <ul style="list-style-type: none"> • In this example, the EEM policy with a job ID of 2 is cleared from the pending queue.
Step 4	show event manager policy pending Example: Device# show event manager policy pending	Displays all the pending EEM policies except the policy cleared in Step 3. Note Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.

Examples

The following example shows how to clear the EEM policy with a job ID of 2 that was pending execution. The **show** commands are used to display the policies that are pending before and after the policy is cleared.

```

Device# show event manager policy pending
no. job id status time of event          event type      name
1   1      pend  Thu Sep 7  02:54:04 2006  syslog         applet: one
2   2      pend  Thu Sep 7  02:54:04 2006  syslog         applet: two
3   3      pend  Thu Sep 7  02:54:04 2006  syslog         applet: three

Device# event manager scheduler clear policy 2
Device# show event manager policy pending

no. job id status time of event          event type      name
1   1      pend  Thu Sep 7  02:54:04 2006  syslog         applet: one
3   3      pend  Thu Sep 7  02:54:04 2006  syslog         applet: three

```

Modifying the Scheduling Parameters of EEM Policy Events or Event Queues

To modify the scheduling parameters of the EEM policies, perform this task. The **show event manager policy pending** command displays policies that are assigned to the B or default class. All the currently pending policies are then changed to class A. After the configuration modification, the **show event manager policy pending** command shows all policies assigned as class A.

SUMMARY STEPS

1. enable
2. show event manager policy pending
3. event manager scheduler modify {all | policy *job-id* | queue-type {applet | call-home | axp | script} | class *class-options*} [queue-priority {high | last | low | normal}][processor {rp_primary | rp_standby}]
4. show event manager policy pending

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager policy pending Example: Device# show event manager policy pending	Displays the pending EEM policies. Note Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.
Step 3	event manager scheduler modify {all policy <i>job-id</i> queue-type {applet call-home axp script} class <i>class-options</i>} [queue-priority {high last low normal}][processor {rp_primary rp_standby}] Example: Device# event manager scheduler modify all class A	Modifies the scheduling parameters of the EEM policies. <ul style="list-style-type: none"> • In this example, all currently pending EEM policies are assigned to class A.

	Command or Action	Purpose
Step 4	<p>show event manager policy pending</p> <p>Example:</p> <pre>Device# show event manager policy pending</pre>	<p>Displays the EEM policies modified in Step 3 along with other pending policies.</p> <p>Note Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.</p>

Examples

The following example shows how to modify the scheduling parameters of the EEM policies. In this example, the **show event manager policy pending** command displays policies that are assigned to the B or default class. All the currently pending policies are then changed to class A. After the configuration modification, the **show event manager policy pending** command verifies that all policies are now assigned as class A.

```
Device# show event manager policy pending
no. class status time of event event type name
1 default pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 default pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 B pend Thu Sep 7 02:54:04 2006 syslog applet: three

Device# event manager scheduler modify all class A
Device# show event manager policy pending

no. class status time of event event type name
1 A pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 A pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 A pend Thu Sep 7 02:54:04 2006 syslog applet: three
```

Verifying Class-Based Active EEM Policies

To verify the active or the running EEM policies, use the **show event manager policy active** command.

SUMMARY STEPS

1. **show event manager policy active** [**queue-type** {**applet** | **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

DETAILED STEPS

show event manager policy active [**queue-type** {**applet** | **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

This command displays only the running EEM policies. This command includes **class**, **detailed** and **queue-type** optional keywords. The following is sample output from this command:

Example:

```
Device# show event manager policy active
no. job id p s status time of event event type name
1 12598 N A running Mon Oct29 20:49:37 2007 timer watchdog loop.tcl
```

```

2 12609 N A running Mon Oct29 20:49:42 2007 timer watchdog loop.tcl
3 12620 N A running Mon Oct29 20:49:46 2007 timer watchdog loop.tcl
4 12650 N A running Mon Oct29 20:49:59 2007 timer watchdog loop.tcl
5 12842 N A running Mon Oct29 20:51:13 2007 timer watchdog loop.tcl
default class - 6 applet events
no. job id p s status time of event event type name
1 15852 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
2 15853 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
3 15854 N A running Mon Oct29 21:11:10 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
4 15855 N A running Mon Oct29 21:11:10 2007 timer watchdog WDOG_SYSLG_CNTR_TRACK_INTF_APPL
5 15856 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
6 15858 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL

```

Verifying Class-Based Active EEM Policies

To verify the active or the running EEM policies, use the **show event manager policy active** command.

SUMMARY STEPS

1. **show event manager policy active** [*queue-type* {*applet* | *call-home* | *axp* | *script*}] *class class-options* | *detailed*]

DETAILED STEPS

```
show event manager policy active [queue-type {applet | call-home | axp | script}] class class-options | detailed]
```

This command displays only the running EEM policies. This command includes **class**, **detailed** and **queue-type** optional keywords. The following is sample output from this command:

Example:

```

Device# show event manager policy active
no. job id p s status time of event event type name
1 12598 N A running Mon Oct29 20:49:37 2007 timer watchdog loop.tcl
2 12609 N A running Mon Oct29 20:49:42 2007 timer watchdog loop.tcl
3 12620 N A running Mon Oct29 20:49:46 2007 timer watchdog loop.tcl
4 12650 N A running Mon Oct29 20:49:59 2007 timer watchdog loop.tcl
5 12842 N A running Mon Oct29 20:51:13 2007 timer watchdog loop.tcl
default class - 6 applet events
no. job id p s status time of event event type name
1 15852 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
2 15853 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
3 15854 N A running Mon Oct29 21:11:10 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
4 15855 N A running Mon Oct29 21:11:10 2007 timer watchdog WDOG_SYSLG_CNTR_TRACK_INTF_APPL
5 15856 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
6 15858 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL

```

Verifying Pending EEM Policies

To verify the EEM policies that are pending for execution, use the **show event manager policy pending** command. Use the optional keywords to specify EEM class-based scheduling options.

SUMMARY STEPS

1. **show event manager policy pending** [*queue-type* {*applet*|*call-home* | *axp* | *script*}] *class class-options* | **detailed**]

DETAILED STEPS

show event manager policy pending [*queue-type* {*applet*|*call-home* | *axp* | *script*}] *class class-options* | **detailed**]

This command displays only the pending policies. This command includes **class**, **detailed** and **queue-type** optional keywords. The following is sample output from this command:

Example:

```
Device# show event manager policy pending
no. job id p s status time of event event type name
1 12851 N A pend Mon Oct29 20:51:18 2007 timer watchdog loop.tcl
2 12868 N A pend Mon Oct29 20:51:24 2007 timer watchdog loop.tcl
3 12873 N A pend Mon Oct29 20:51:27 2007 timer watchdog loop.tcl
4 12907 N A pend Mon Oct29 20:51:41 2007 timer watchdog loop.tcl
5 13100 N A pend Mon Oct29 20:52:55 2007 timer watchdog loop.tcl
```

Configuring EEM Applet (Interactive CLI) Support

The synchronous applets are enhanced to support interaction with the local console (tty) using two commands, **action gets** and **action puts**, and these commands allow users to enter and display input directly on the console. The output for synchronous applets will bypass the system logger. The local console will be opened by the applets and serviced by the corresponding synchronous Event Detector pty. Synchronous output will be directed to the opened console.

Reading and Writing Input from the Active Console for Synchronous EEM Applets

Use the following tasks to implement EEM applet interactive CLI support:

Reading Input from the Active Console

When a synchronous policy is triggered, the related console is stored in the publish information specification. The policy director will query this information in an event_reqinfo call, and store the given console information for use by the **action gets** command.

The **action gets** command reads a line of the input from the active console and stores the input in the variable. The trailing new line will not be returned.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event none**
5. **action** *label* **gets** *variable*
6. **action** *label* **syslog** [*priority priority-level*] **msg** *msg-text*

7. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet action	Registers the applet with the EEM and enters applet configuration mode.
Step 4	event none Example: Device(config-applet)# event none	Specifies that an EEM policy is to be registered with the EEM and can be run manually.
Step 5	action <i>label</i> gets <i>variable</i> Example: Device(config-applet)# action label2 gets input	Gets input from the local console in a synchronous applet and stores the value in the given variable when an EEM applet is triggered.
Step 6	action <i>label</i> syslog [<i>priority</i> <i>priority-level</i>] msg <i>msg-text</i> Example: Device(config-applet)# action label3 syslog msg "Input entered was \"\$input\""	Specifies the action to be taken when an EEM applet is triggered. <ul style="list-style-type: none">• In this example, the action to be taken is to write the value of the variable specified in Step 5, to syslog.
Step 7	exit Example: Device(config-applet)# exit	Exits applet configuration mode and returns to privileged EXEC mode.

Example

The following example shows how to get the input from the local tty in a synchronous applet and store the value

```
Device(config)# event manager applet action
Device(config-applet)# event none
```



```
Device(config-applet)# action label2 gets input
Device(config-applet)# action label3 syslog msg "Input entered was \"${input}\""
```

Writing Input to the Active Console

When a synchronous policy is triggered, the related console is stored in the publish information specification. The policy director will query this information in an event_reqinfo call, and store the given console information for use by the **action puts** command.

The **action puts** command will write the string to the active console. A new line will be displayed unless the **newline** keyword is specified. The output from the **action puts** command for a synchronous applet is displayed directly to the console, bypassing the system logger. The output of the **action puts** command for an asynchronous applet is directed to the system logger.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event none**
5. **action** *label* **regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]]
6. **action** *label* **puts** [**newline**] *string*
7. **exit**
8. **event manager run** *applet-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet action	Registers the applet with the EEM and enters applet configuration mode.
Step 4	event none Example: Device(config-applet)# event none	Specifies that an EEM policy is to be registered with the EEM and can be run manually.

	Command or Action	Purpose
Step 5	<p>action <i>label</i> regexp <i>string-pattern</i> <i>string-input</i> [<i>string-match</i> [<i>string-submatch1</i>] [<i>string-submatch2</i>] [<i>string-submatch3</i>]]</p> <p>Example:</p> <pre>Device(config-applet)# action 1 regexp "(.*) (.*) (.*)" "one two three" _match _sub1</pre>	Specifies the action to match the regular expression pattern on an input string when an EEM applet is triggered.
Step 6	<p>action <i>label</i> puts [newline] <i>string</i></p> <p>Example:</p> <pre>Device(config-applet)# action 2 puts "match is \$_match"</pre>	<p>Specifies the action of printing data directly to the local console when an EEM applet is triggered.</p> <ul style="list-style-type: none"> The newline keyword is optional and is used to suppress the display of the new line character.
Step 7	<p>exit</p> <p>Example:</p> <pre>Device(config-applet)# exit</pre>	Exits applet configuration mode and returns to privileged EXEC mode.
Step 8	<p>event manager run <i>applet-name</i></p> <p>Example:</p> <pre>Device# event manager run action</pre>	<p>Manually runs a registered EEM policy.</p> <ul style="list-style-type: none"> In this example, the policy registered in Step 3 is triggered and the associated actions specified in Step 5 and Step 6 are executed.

Example

The following example shows how the **action puts** command prints data directly to the local console:

```
Device(config-applet)# event manager applet puts
Device(config-applet)# event none
Device(config-applet)# action 1 regexp "(.*) (.*) (.*)" "one two three" _match _sub1
Device(config-applet)# action 2 puts "match is $_match"
Device(config-applet)# action 3 puts "submatch 1 is $_sub1"
Device# event manager run puts
match is one two three
submatch 1 is one
```

Configuring SNMP Library Extensions

Depending on your release, the SNMP Library Extensions feature allows you to perform the following configurations.

Prerequisites

To use this feature, you must be running Cisco IOS Release 12.4(22)T or a later release.

SNMP Get and Set Operations

The SNMP Library Extensions feature extends the EEM applet **action info** and Tcl **sys_reqinfo_snmp** commands to include functionality for SNMP get-one, get-next, getid and set-any operations.

SNMP Get Operation

The SNMP event manager performs the SNMP get operation to retrieve one or more variables for the managed objects. Using the **action info type snmp oid get-type** and **action info type snmp getid** commands, you can configure the SNMP event manager to send an SNMP get request by specifying the variables to retrieve, and the IP address of the agent.

For example, if you want to retrieve the variable with the OID value of 1.3.6.1.2.1.1.1, you should specify the variable value, that is 1.3.6.1.2.1.1.1. If the specified values do not match, a trap will be generated and an error message will be written to the syslog history.

The **action info type snmp oid get-type** command specifies the type of the get operation to be performed. To retrieve the exact variable, the get operation type should be specified as **exact**. To retrieve a lexicographical successor of the specified OID value, the get operation type should be set to **next**.

The table below shows the built-in variables, in which the values retrieved from SNMP get operation are stored.

Table 181: Built-in Variables for action info type snmp oid Command

Built-in Variable	Description
_info_snmp_oid	The SNMP object ID.
_info_snmp_value	The value string of the associated SNMP data element.

GetID Operation

The **action info type snmp getid** command retrieves the following variables from the SNMP entity:

- sysDescr.0
- sysObjectID.0
- sysUpTime.0
- sysContact.0
- sysName.0
- sysLocation.0

The table below shows the built-in variables, in which the values retrieved from the SNMP getID operation are stored.

Table 182: Built-in Variables for action info type snmp getid Command

Built-in Variable	Description
_info_snmp_syslocation_oid	The OID value of the sysLocation variable.
_info_snmp_syslocation_value	The value string for the sysLocation variable.

Built-in Variable	Description
<code>_info_snmp_sysdescr_oid</code>	The OID value of the sysDescr variable.
<code>_info_snmp_sysdescr_value</code>	The value string for the sysDescr variable.
<code>_info_snmp_sysobjectid_oid</code>	The OID value of the sysObjectID variable.
<code>_info_snmp_sysobjectid_value</code>	The value string for the sysObjectID variable.
<code>_info_snmp_sysuptime_oid</code>	The OID value of the sysUptime variable.
<code>_info_snmp_sysuptime_value</code>	The value string for the sysUptime variable.
<code>_info_snmp_syscontact_oid</code>	The OID value of the sysContact variable.
<code>_info_snmp_syscontact_value</code>	The value string for the sysContact variable.

The get operation requests can be sent to both local and remote hosts.

SNMP Set Operation

All SNMP variables are assigned a default value in the MIB view. The SNMP event manager can modify the value of these MIB variables through set operation. The set operation can be performed only on the system that allows read-write access.

To perform a set operation, you must specify the type of the variable and the value associated with it.

The table below shows the valid OID types and values for each OID type.

Table 183: OID Type and Value for Set Operation

OID Type	Description
counter32	A 32-bit number with a minimum value of 0. Value in the range from 0 to 4294967295 is valid.
gauge	A 32-bit number with a minimum value of 0. Integer value in the range from 0 to 4294967295 is valid.
integer	A 32-bit number used to specify a numbered object type. Integer value in the range from 0 to 4294967295 is valid.
ipv4	IP version 4 address. IPv4 address in dotted decimal notation.
octet string	An octet string in hexadecimal notation used to represent a string of octets.
string	An octet string in text notation used to represent a string of octets.
unsigned32	A 32-bit number used to represent decimal value. Value in the range from 0 to 4294967295 is valid.

The set operation can be carried out on both local and remote hosts.

SNMP Traps and Inform Requests

Traps are SNMP notifications that alert the SNMP manager or the NMS to a network condition.

SNMP inform requests refer to the SNMP notifications that alert the SNMP manager to a network condition and request for confirmation of receipt from the SNMP manager.

An SNMP event occurs when SNMP MIB object ID values are sampled, or when the SNMP counter crosses a defined threshold. If the notifications are enabled and configured for such events, the SNMP traps or inform messages generated. An SNMP notification event is triggered when an SNMP trap or inform message is received by the event manager server.

To send an SNMP trap or inform message when an Embedded Event Manager (EEM) applet is triggered, the **action info type snmp trap** and **action info type snmp inform** commands are used. The CISCO-EMBEDDED-EVENT-MGR-MIB.mib is used to define the trap and inform messages.

Configuring EEM Applet for SNMP Get and Set Operations

While registering a policy with the event manager server, the actions associated with an SNMP event can be configured.

Perform this task to configure EEM applet for SNMP set and get operations.

Before you begin

- SNMP event manager must be configured using the **snmp-server manager** command.
- The SNMP community string should be set by using the **snmp-server community** command to enable access to the SNMP entity.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. Do one of the following:
 - **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value* [**exit-comb** | **and**] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
5. **action label info type snmp oid** *oid-value* **get-type** {**exact** | **next**} [**community** *community-string*] [**ipaddr** *ip-address*]
6. **action label info type snmp oid** *oid-value* **set-type** *oid-type* *oid-type-value* **community** *community-string* [**ipaddr** *ip-address*]
7. **action label info type snmp getid** *oid-value* [**community** *community-string*] [**ipaddr** *ip-address*]
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet snmp	Registers the applet with the event manager server and enters applet configuration mode.
Step 4	Do one of the following: <ul style="list-style-type: none"> • event snmp oid <i>oid-value</i> get-type {exact next} entry-op <i>operator</i> entry-val <i>entry-value</i> [exit-comb and] [exit-op <i>operator</i>] [exit-val <i>exit-value</i>] [exit-time <i>exit-time-value</i>] poll-interval <i>poll-int-value</i> Example: Device(config-applet)# event snmp oid Example: 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact Example: entry-op lt entry-val 5120000 poll-interval 90	Specifies the event criteria that cause the EEM applet to run. <ul style="list-style-type: none"> • In this example, an EEM event is triggered when free memory falls below the value of 5120000. • Exit criteria are optional, and if not specified, event monitoring is reenabled immediately.
Step 5	action label info type snmp oid <i>oid-value</i> get-type { exact next } [community <i>community-string</i>] [ipaddr <i>ip-address</i>] Example: Device(config-applet)# action 1.3 info type Example: snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type Example: exact community public ipaddr 172.17.16.69	Specifies the type of get operation to perform. <ul style="list-style-type: none"> • In this example, the type of get operation is specified as exact and community string is specified as public.
Step 6	action label info type snmp oid <i>oid-value</i> set-type <i>oid-type</i> <i>oid-type-value</i> community <i>community-string</i> [ipaddr <i>ip-address</i>] Example: Device(config-applet)# action 1.4 info type	(Optional) Specifies the variable to be set. <ul style="list-style-type: none"> • In this example, the sysName.0 variable is specified for the set operation and community string is specified as rw.

	Command or Action	Purpose
	<p>Example:</p> <pre>snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 set-type</pre> <p>Example:</p> <pre>integer 42220 sysName.0 community rw ipaddr</pre> <p>Example:</p> <pre>172.17.16.69</pre>	<p>Note For set operation, you must specify the SNMP community string.</p>
Step 7	<p>action label info type snmp getid <i>oid-value</i> [community <i>community-string</i>] [ipaddr <i>ip-address</i>]</p> <p>Example:</p> <pre>Device(config-applet)# action 1.3 info type</pre> <p>Example:</p> <pre>snmp getid community public ipaddr 172.17.16.69</pre>	(Optional) Specifies if the individual variables should be retrieved by the getid operation.
Step 8	<p>exit</p> <p>Example:</p> <pre>Device(config)# exit</pre>	Exits global configuration mode and returns to privileged EXEC mode.

Configuring EEM Applet for SNMP OID Notifications

Perform this task to configure SNMP notifications.

Before you begin

- SNMP event manager must be configured using the **snmp-server manager** command and SNMP agents must be configured to send and receive SNMP traps generated for an EEM policy.
- SNMP traps and informs must be enabled by using the **snmp-server enable traps event-manager** and **snmp-server enable traps** commands, to allow traps and inform requests to be sent from the device to the event manager server.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. Do one of the following:
 - **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value* [**exit-comb** | **and**] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*

5. **action label info type snmp var** *variable-name* **oid** *oid-value* *oid-type* *oid-type-value*
6. **action label info type snmp trap enterprise-oid** *enterprise-oid-value* **generic-trapnum** *generic-trap-number* **specific-trapnum** *specific-trap-number* **trap-oid** *trap-oid-value* **trap-var** *trap-variable*
7. **action label info type snmp inform trap-oid** *trap-oid-value* **trap-var** *trap-variable* **community** *community-string* **ipaddr** *ip-address*
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet snmp	Registers the applet with the event manager server and enters applet configuration mode.
Step 4	Do one of the following: <ul style="list-style-type: none"> • event snmp oid <i>oid-value</i> get-type {exact next} entry-op <i>operator</i> entry-val <i>entry-value</i> [exit-comb and] [exit-op <i>operator</i>] [exit-val <i>exit-value</i>] [exit-time <i>exit-time-value</i>] poll-interval <i>poll-int-value</i> Example: Device(config-applet)# event snmp oid Example: 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact Example: entry-op lt entry-val 5120000 poll-interval 90	Specifies the event criteria that cause the EEM applet to run. <ul style="list-style-type: none"> • In this example, an EEM event is triggered when free memory falls below the value of 5120000. • Exit criteria are optional, and if not specified, event monitoring is reenabled immediately.
Step 5	action label info type snmp var <i>variable-name</i> oid <i>oid-value</i> <i>oid-type</i> <i>oid-type-value</i> Example: Device(config-applet)# action 1.3 info type Example:	Specifies the instance of a managed object and its value. <ul style="list-style-type: none"> • In this example, the sysDescr.0 variable is used.

	Command or Action	Purpose
	<pre>snmp var sysDescr.0 oid</pre> <p>Example:</p> <pre>1.3.6.1.4.1.9.9.48.1.1.1.6.1 integer 4220</pre>	
Step 6	<p>action label info type snmp trap enterprise-oid <i>enterprise-oid-value generic-trapnum</i> <i>generic-trap-number specific-trapnum</i> <i>specific-trap-number trap-oid trap-oid-value trap-var</i> <i>trap-variable</i></p> <p>Example:</p> <pre>Device(config-applet)# action 1.4 info type</pre> <p>Example:</p> <pre>snmp trap enterprise-oid 1.3.6.1.4.1.1</pre> <p>Example:</p> <pre>generic-trapnum 4 specific-trapnum 7 trap-oid</pre> <p>Example:</p> <pre>1.3.6.1.4.1.1.226.0.2.1 trap-var sysUpTime.0</pre>	<p>Generates an SNMP trap when the EEM applet is triggered.</p> <ul style="list-style-type: none"> In this example, the authenticationFailure trap is generated. <p>Note The specific trap number refers to the enterprise-specific trap, which is generated when an enterprise event occurs. If the generic trap number is not set to 6, the specific trap number you specify will be used to generate traps.</p>
Step 7	<p>action label info type snmp inform trap-oid <i>trap-oid-value trap-var trap-variable community</i> <i>community-string ipaddr ip-address</i></p> <p>Example:</p> <pre>Device(config-applet)# action 1.4 info type</pre> <p>Example:</p> <pre>snmp inform trap-oid 1.3.6.1.4.1.1.226.0.2.1</pre> <p>Example:</p> <pre>trap-var sysUpTime.0 community public ipaddr</pre> <p>Example:</p> <pre>172.69.16.2</pre>	<p>Generates an SNMP inform request when the EEM applet is triggered.</p> <ul style="list-style-type: none"> In this example, the inform request is generated for the sysUpTime.0 variable.
Step 8	<p>exit</p> <p>Example:</p> <pre>Device(config)# exit</pre>	<p>Exits global configuration mode and returns to privileged mode.</p>

Configuring Variable Logic for EEM Applets

The Variable Logic for EEM Applets feature adds the ability to apply conditional logic within EEM applets. Before variable logic is introduced, applets have a linear structure where each action is executed in the order in which they are configured when the event is triggered. Conditional logic introduces a control structure that can change the flow of actions within applets depending on conditional expressions. Each control structure can contain a list of applet actions including looping and if/else actions which determine if the structure is executed or not.

The information in applet configuration mode is presented as background to set the context for the action commands.

To provide a consistent user interface between the Tool Command Language (Tcl) and the applet (CLI) based EEM policies, the following criteria are followed:

- Event specification criteria are written in Tcl in the Tcl based implementation.
- Event specification data is written using the CLI applet submode configuration statements in the applet-based implementation.

Applet configuration mode is entered using the event manager applet command. In applet configuration mode the config prompt changes to (config-applet)#. In applet configuration mode two types of config statements are supported:

- event - used to specify the event criteria to cause this applet to run.
- action - used to specify a built-in action to perform.

Multiple **action** applet config commands are allowed within an applet configuration. If no **action** applet config command is present, a warning is displayed, upon exit, stating no statements are associated with this applet. When no statements are associated with this applet, events get triggered but no action is taken. If no commands are specified in applet configuration mode, the applet will be removed upon exit. The exit applet config command is used to exit from applet configuration mode.

Depending on your release, the Variable Logic for EEM Applets feature allows you to perform the following configurations.

Prerequisites

To use this feature, you must be running Cisco IOS Release 12.4(22)T or a later release.

Configuring Variable Logic for EEM Applets

EEM 3.0 adds new applet action commands to permit simple variable logic within applets.

To configure the variable logic using action commands perform the following tasks.

Specifying a Loop of Conditional Blocks

To specify a loop of a conditional block when an EEM applet is triggered, perform this task. In this task, a conditional loop is set to check if the value of the variable is less than 10. If the value of the variable is less than 10, then the message 'i is \$_i' is written to the syslog.



Note Depending on your release, the **set** (EEM) command is replaced by the **action set** command. See the **action label set** command for more information. If the set (EEM) command is entered in certain releases, the IOS parser translates the **set** command to the **action label set** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label set**
5. **action label while** *string_op1 operator string_op2*
6. Add any action as required.
7. **action label end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet condition	Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.
Step 4	action label set Example: Device(config-applet)# action 1.0 set i 2	Sets an action for the event. • In this example, the value of the variable i is set to 2.
Step 5	action label while <i>string_op1 operator string_op2</i> Example: Device(config-applet)# action 2 while \$i lt 10	Specifies a loop of a conditional block. • In this example, a loop is set to check if the value of the variable i is less than 10.
Step 6	Add any action as required. Example:	Performs the action as indicated by the action command. • In this example, the message ‘i is \$_i’ is written to the syslog.

	Command or Action	Purpose
	Device(config-applet)# action 3 syslog msg "i is \$i"	
Step 7	action label end Example: Device(config-applet)# action 3 end	Exits from the running action.

Specifying if else Conditional Blocks

To specify the beginning of an if conditional statement followed by an else conditional statement, perform this task. The if or else conditional statements can be used in conjunction with each other or separately. In this task, the value of a variable is set to 5. An if conditional block is then specified to check if the value of the variable is less than 10. Provided the if conditional block is satisfied, an action command to output the message 'x is less than 10' is specified.

Following the if conditional block, an else conditional block is specified. Provided the if conditional block is not satisfied, an action command to output the message 'x is greater than 10' is specified.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label set** *variable-name variable-value*
5. **action label if** [*stringop1*] {**eq** | **gt** | **ge** | **lt** | **le** | **ne**} [*stringop2*]
6. Add any action as required.
7. **action label else**
8. Add any action as required.
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example:	Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.

	Command or Action	Purpose
	<code>Device(config)# event manager applet ifcondition</code>	
Step 4	action <i>label</i> set <i>variable-name</i> <i>variable-value</i> Example: <code>Device(config-applet)# action 1.0 set x 5</code>	Sets an action for the event. <ul style="list-style-type: none"> • In this example, the value of the variable x is set to 5.
Step 5	action <i>label</i> if [<i>stringop1</i>] { eq gt ge lt le ne } [<i>stringop2</i>] Example: <code>Device(config-applet)# action 2.0 if \$x lt 10</code>	Specifies an if conditional statement. <ul style="list-style-type: none"> • In this example, an if conditional statement to check if the value of the variable is less than 10.
Step 6	Add any action as required. Example: <code>Device(config-applet)# action 3.0 puts "\$x is less than 10"</code>	Performs the action as indicated by the action command. <ul style="list-style-type: none"> • In this example, the message '5 is less than 10' is displayed on the screen.
Step 7	action <i>label</i> else Example: <code>Device(config-applet)# action 4.0 else</code>	Specifies an else conditional statement
Step 8	Add any action as required. Example: <code>Device(config-applet)# action 5.0</code>	Performs the action as indicated by the action command. <ul style="list-style-type: none"> • In this example, the message '5 is greater than 10' is displayed on the screen.
Step 9	end Example: <code>Device(config-applet)# end</code>	Exits from the running action.

Specifying foreach Iterating Statements

To specify a conditional statement that iterates over an input string using the delimiter as a tokenizing pattern, perform this task. The foreach iteration statement is used to iterate through a collection to get the desired information. The delimiter is a regular expression pattern string. The token found in each iteration is assigned to the given iterator variable. All arithmetic calculations are performed as long integers with out any checks for overflow. In this task, the value of the variable x is set to 5. An iteration statement is set to run through the input string red, blue, green, orange. For every element in the input string, a corresponding message is displayed on the screen.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **event manager applet** *applet-name*
4. **action label foreach** [*string-iterator*] [*string-input*] [*string-delimiter*]
5. Specify any action command
6. **action label end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet iteration	Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.
Step 4	action label foreach [<i>string-iterator</i>] [<i>string-input</i>] [<i>string-delimiter</i>] Example: Device(config-applet)# action 2.0 foreach iterator "red blue green orange"	Iterates over an input string using the delimiter as a tokenizing pattern. <ul style="list-style-type: none"> • In this example, the iteration is run through the elements of the input string - red, blue, green and orange.
Step 5	Specify any action command Example: Device(config-applet)# action 3.0 puts "Iterator is \$iterator"	Performs the action as indicated by the action command. <ul style="list-style-type: none"> • In this example, the following message is displayed on the screen: Iterator is red Iterator is blue Iterator is green Iterator is orange
Step 6	action label end Example: Device(config-applet)# action 4.0 end	Exits from the running action.

Using Regular Expressions

To match a regular expression pattern with an input string, perform this task. Using regular expressions, you can specify the rules for a set of possible strings to be matched.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet regexp	Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.
Step 4	action label regexp <i>string-pattern string-input</i> [<i>string-match</i> [<i>string-submatch1</i>] [<i>string-submatch2</i>] [<i>string-submatch3</i>]] Example: Device(config-applet)# action 2.0 regexp "(.*) (.*)(.*) "red blue green" _match _sub1	Specifies an expression pattern to match with an input string. <ul style="list-style-type: none"> • In this example, an input string of 'red blue green' is specified. When the expression pattern matches the input string, the entire result red blue green is stored in the variable _match and the submatch red is stored in the variable _sub1.

Incrementing the Values of Variables

To increment the value of variables, perform this task. In this task, the value of a variable is set to 20 and then the value is incremented by 12.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label set**
5. **action label increment** *variable-name long-integer*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	event manager applet <i>applet-name</i> Example: Device(config)# event manager applet increment	Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.
Step 4	action label set Example: Device(config-applet)# action 1.0 set varname 20	Sets an action for the event. <ul style="list-style-type: none"> • In this example, the value of the variable is set to 20.
Step 5	action label increment <i>variable-name long-integer</i> Example: Device(config-applet)# action 2.0 increment varname 12	Increments the value of variable by the specified long integer. <ul style="list-style-type: none"> • In this example, the value of the variable is incremented by 12.

Configuring Event SNMP Object

Perform this task to register the Simple Network Management Protocol (SNMP) object event for an Embedded Event Manager (EEM) applet that is run by sampling SNMP object.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event snmp-object oid** *oid-value* **type** *value* **sync** {yes | no} **skip** {yes | no} **istable** {yes | no} [default *seconds*] [maxrun *maxruntime-number*]

5. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>event manager applet <i>applet-name</i></p> <p>Example:</p> <pre>Device(config)# event manager applet manual-policy</pre>	<p>Registers the applet with the Embedded Event Manager and enters applet configuration mode.</p>
Step 4	<p>event snmp-object oid <i>oid-value</i> type <i>value</i> sync {yes no} skip {yes no} istable {yes no} [default <i>seconds</i>] [maxrun <i>maxruntime-number</i>]</p> <p>Example:</p> <pre>Device(config-applet)# event snmp-object oid 1.9.9.9 type gauge sync yes</pre> <p>Example:</p> <pre>action 1 syslog msg "oid = \$_snmp_oid"</pre> <p>Example:</p> <pre>action 2 syslog msg "request = \$_snmp_request"</pre> <p>Example:</p> <pre>action 3 syslog msg "request_type = \$_snmp_request_type"</pre>	<p>Registers the Simple Network Management Protocol (SNMP) object event for an Embedded Event Manager (EEM) applet to intercept SNMP GET and SET requests for an object.</p> <p>The default for this command is that it is not configured. If this command is configured the defaults are the same as in the description of the syntax options,</p> <ul style="list-style-type: none"> • The oid keyword specifies the SNMP object identifier (object ID). • The <i>oid-value</i> argument can be the Object ID value of the data element, in SNMP dotted notation. An OID is defined as a type in the associated MIB, CISCO-EMBEDDED-EVENT-MGR-MIB, and each type has an object value. • The istable keyword specifies whether the OID is an SNMP table. • The sync keyword specifies that the applet is to run in synchronous mode. The return code from the applet indicates whether to reply to the SNMP request. The description for code 0 is “do not reply to the request” and the description for code 1 is “reply to the request”. When the return code from the applet replies to the request, a value is specified in the applet for the object using action snmp-object-value command. • The type keyword specifies the type of object. • The <i>value</i> argument is the value of the object.

	Command or Action	Purpose
		<ul style="list-style-type: none"> The skip keyword specifies whether to skip CLI command execution. The default keyword specifies the time to process the SET or GET request normally by the applet. If the default keyword is not specified, the default time period is set to 30 seconds. The <i>milliseconds</i> argument is the time period during which the SNMP Object event detector waits for the policy to exit. The maxrun keyword specifies the maximum runtime of the applet. If the maxrun keyword is specified, the <i>maxruntime-number</i> value must be specified. If the maxrun keyword is not specified, the default applet run time is 20 seconds. The <i>milliseconds</i> argument is the maximum runtime of the applet in milliseconds. If the argument is not specified, the default 20-second run-time limit is used.
Step 5	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Disabling AAA Authorization

Perform this task to allow EEM policies to bypass AAA authorization when triggered.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name* [**authorization bypass**] [**class** *class-options*] [**trap**]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	<code>Device# configure terminal</code>	
Step 3	<p>event manager applet <i>applet-name</i> [authorization bypass] [class <i>class-options</i>] [trap]</p> <p>Example:</p> <pre>Device(config)# event manager applet one class A authorization bypass</pre>	Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.
Step 4	<p>exit</p> <p>Example:</p> <pre>Device(config-aaplet)# exit</pre>	Exits device configuration applet mode and returns to privileged EXEC mode.

Configuring Description of an Embedded Event Manager Applet

Perform this task to describe an EEM applet. The description of an applet can be added in any order, before or after any other applet configuration. Configuring a new description for an applet that already has a description overwrites the current description. An applet description is optional.

Perform this task to configure a new description for an applet.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **description** *line*
5. **event syslog pattern** *regular-expression*
6. **action** *label* **syslog msg** *msg-text*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	event manager applet <i>applet-name</i> Example: <pre>Device(config)# event manager applet increment</pre>	Registers the applet with the EEM and enters applet configuration mode.
Step 4	description <i>line</i> Example: <pre>Device(config-applet)# description "This applet looks for the word count in syslog messages"</pre>	Adds or modifies the description of an EEM applet that is run by sampling Simple Network Management Protocol (SNMP).
Step 5	event syslog pattern <i>regular-expression</i> Example: <pre>Device(config-applet)# event syslog pattern "count"</pre>	Specifies the event criteria for an Embedded Event Manager (EEM) applet that is run by matching syslog messages.
Step 6	action <i>label</i> syslog msg <i>msg-text</i> Example: <pre>Device(config-applet)# action 1 syslog msg hi</pre>	Specifies the action to be taken when an EEM applet is triggered. <ul style="list-style-type: none"> • In this example, the action taken is to write a message to syslog. • The <i>msg-text</i> argument can be character text, an environment variable, or a combination of the two.
Step 7	end Example: <pre>Device(config-applet)# end</pre>	Exits applet configuration mode and returns to privileged EXEC mode.

Configuration Examples for Writing Embedded Event Manager Policies Using Tcl

Embedded Event Manager Applet Configuration Examples

The following examples show how to create an EEM applet for some of the EEM event detectors. These examples follow steps outlined in the [Registering and Defining an Embedded Event Manager Applet, on page 1769](#).

Application-Specific Event Detector

The following example shows how a policy named EventPublish_A runs every 20 seconds and publishes an event type numbered 1 to an EEM subsystem numbered 798. The subsystem value of 798 specifies that a publish event has occurred from an EEM policy. A second policy named EventPublish_B is registered to run

when the EEM event type 1 occurs with subsystem 798. When the EventPublish_B policy runs, it sends a message to syslog containing data passed as an argument from the EventPublish_A policy.

```
event manager applet EventPublish_A
  event timer watchdog time 20.0
  action 1.0 syslog msg "Applet EventPublish_A"
  action 2.0 publish-event sub-system 798 type 1 arg1 twenty
  exit
event manager applet EventPublish_B
  event application sub-system 798 type 1
  action 1.0 syslog msg "Applet EventPublish_B arg1 $_application_data1"
```

CLI Event Detector

The following example shows how to specify an EEM applet to run when the Cisco IOS **write memory** CLI command is run. The applet provides a notification that this event has occurred via a syslog message. In the example, the **sync** keyword is configured with the **yes** argument, and this means that the event detector is notified when this policy completes running. The exit status of the policy determines whether the CLI command will be executed. In this example, the policy exit status is set to one and the CLI command runs.

```
event manager applet cli-match
  event cli pattern "write mem.*" sync yes
  action 1.0 syslog msg "$_cli_msg Command Executed"
  set 2.0 _exit_status 1
```

The following example shows an applet which matches the **cli pattern** with the **test** argument. When **show access-list test** is entered, the CLI event detector matches the test argument, and the applet is triggered. The **debug event manager detector cli** output is added to show **num_matches** is set to one.

```
!
event manager applet EEM-PIPE-TEST
  event cli pattern "test" sync yes
  action 1.0 syslog msg "Pattern matched!"
!
*Aug 23 23:19:59.827: check_eem_cli_policy_handler: command_string=show access-lists test
*Aug 23 23:19:59.827: check_eem_cli_policy_handler: num_matches = 1, response_code = 4
*Aug 23 23:19:59.843: %HA_EM-6-LOG: EEM-PIPE-TEST: Pattern matched!
```



Note The functionality provided in the CLI event detector only allows a regular expression pattern match on a valid IOS CLI command itself. This does not include text after a pipe (|) character when redirection is used.

The following example shows that when **show version | include test** is entered, the applet fails to trigger because the CLI event detector does not match on characters entered after the pipe (|) character and the **debug event manager detector cli** output shows **num_matches** is set to zero.

```
*Aug 23 23:20:16.827: check_eem_cli_policy_handler: command_string=show version
*Aug 23 23:20:16.827: check_eem_cli_policy_handler: num_matches = 0, response_code = 1
```

Counter Event Detector and Timer Event Detector

The following example shows that the EventCounter_A policy is configured to run once a minute and to increment a well-known counter called **critical_errors**. A second policy--EventCounter_B--is registered to be triggered when the well-known counter called **critical_errors** exceeds a threshold of 3. When the EventCounter_B policy runs, it resets the counter to 0.

```

event manager applet EventCounter_A
  event timer watchdog time 60.0
  action 1.0 syslog msg "EventCounter_A"
  action 2.0 counter name critical_errors op inc value 1
  exit
event manager applet EventCounter_B
  event counter name critical_errors entry-op gt entry-val 3 exit-op lt exit-val 3
  action 1.0 syslog msg "EventCounter_B"
  action 2.0 counter name critical_errors op set value 0

```

Interface Counter Event Detector

The following example shows how a policy named EventInterface is triggered every time the receive_throttle counter for Fast Ethernet interface 0/0 is incremented by 5. The polling interval to check the counter is specified to run once every 90 seconds.

```

event manager applet EventInterface
  event interface name FastEthernet0/0 parameter receive_throttle entry-op ge entry-val 5
  entry-val-is-increment true poll-interval 90
  action 1.0 syslog msg "Applet EventInterface"

```

Resource Event Detector

The following example shows how to specify event criteria based on an ERM event report for a policy defined to report high CPU usage:

```

event manager applet policy-one
  event resource policy cpu-high
  action 1.0 syslog msg "CPU high at $_resource_current_value percent"

```

RF Event Detector

The RF event detector is only available on networking devices that contain dual Route Processors (RPs). The following example shows how to specify event criteria based on an RF state change notification:

```

event manager applet start-rf
  event rf event rf_prog_initialization
  action 1.0 syslog msg "rf state rf_prog_initialization reached"

```

RPC Event Detector

The RPC event detector allows an outside entity to make a Simple Object Access Protocol (SOAP) request to the device and invokes a defined EEM policy or script. The following example shows how an EEM applet called Event_RPC is being registered to run an EEM script:

```

event manager applet Event_RPC
  event rpc
  action print puts "hello there"

```

The following example shows the format of the SOAP request and reply message:

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://www.cisco.com/eem.xsd">
  <SOAP:Body>
    <run_eemscript>
      <script_name>Event_RPC</script_name>

```

```

        </run_eemscript>
    </SOAP:Body>
</SOAP:Envelope>
]]>]]>
<?xml version="1.0" encoding="UTF-8"?><SOAP:Envelope
xmlns:SOAP="http://www.cisco.com/eem.xsd"><SOAP:Body>
<run_eemscript_response><return_code>0</return_code><output></output></run_eemscript_response></SOAP:Body></SOAP:Envelope>]]>]]>

```

SNMP Event Detector

The following example shows how to specify an EEM applet to run when the CPU usage is greater than 75 percent. When the EEM applet runs, the CLI commands **enable** and **show cpu processes** are run, and an e-mail containing the result of the **show cpu processes** command is sent to an engineer.

```

event manager applet snmpcpuge75
  event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.3.1 get-type exact entry-op ge entry-val 75
  poll-interval 10
  action 1.0 cli command "enable"
  action 2.0 cli command "show process cpu"
  action 3.0 mail server "192.168.1.146" to "engineer@cisco.com" from "devtest@cisco.com"
  subject "B25 PBX Alert" body "$_cli_result"

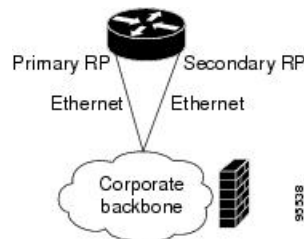
```

The next example is more complex and shows how to configure an EEM applet that causes a switch to the secondary (redundant) Route Processor (RP) when the primary RP runs low on memory.

This example illustrates a method for taking preventative action against a software fault that causes a memory leak. The action taken here is designed to reduce downtime by switching over to a redundant RP when a possible memory leak is detected.

The figure below shows a dual RP device that is running an EEM image. An EEM applet has been registered through the CLI using the **event manager applet** command. The applet will run when the available memory on the primary RP falls below the specified threshold of 5,120,000 bytes. The applet actions are to write a message to syslog that indicates the number of bytes of memory available and to switch to the secondary RP.

Figure 132: Dual RP Topology



The commands used to register the policy are shown below.

```

event manager applet memory-demo
  event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val 5120000
  poll-interval 90
  action 1.0 syslog priority critical msg "Memory exhausted; current available memory is
$_snmp_oid_val bytes"
  action 2.0 force-switchover

```

The registered applet is displayed using the **show event manager policy registered** command:

```

Device# show event manager policy registered
No.  Type      Event Type          Time Registered      Name

```

```

1  applet snmp                               Thu Jan30 05:57:16 2003 memory-demo
   oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
  action 1.0 syslog priority critical msg "Memory exhausted; current available memory is
$_snmp_oid_val bytes"
  action 2.0 force-switchover

```

For the purpose of this example, a memory depletion is forced on the device, and a series of **show memory** commands are executed to watch the memory deplete:

```

Device# show memory
      Head      Total(b)    Used(b)    Free(b)    Lowest(b)  Largest(b)
Processor 53585260 212348444 119523060 92825384 92825384 92365916
Fast      53565260    131080    70360     60720    60720    60668
Device# show memory
      Head      Total(b)    Used(b)    Free(b)    Lowest(b)  Largest(b)
Processor 53585260 212364664 164509492 47855172 47855172 47169340
Fast      53565260    131080    70360     60720    60720    60668
Device# show memory
      Head      Total(b)    Used(b)    Free(b)    Lowest(b)  Largest(b)
Processor 53585260 212369492 179488300 32881192 32881192 32127556
Fast      53565260    131080    70360     60720    60720    60668

```

When the threshold is reached, an EEM event is triggered. The applet named `memory-demo` runs, causing a syslog message to be written to the console and a switch to be made to the secondary RP. The following messages are logged:

```

00:08:31: %HA_EM-2-LOG: memory-demo: Memory exhausted; current available memory is
4484196 bytes
00:08:31: %HA_EM-6-FMS_SWITCH_HARDWARE: fh_io_msg: Policy has requested a hardware
switchover

```

The following is partial output from the **show running-config** command on both the primary RP and the secondary (redundant) RP:

```

redundancy
 mode sso
 .
 .
 !
event manager applet memory-demo
 event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val
5120000 poll-interval 90
  action 1.0 syslog priority critical msg "Memory exhausted; current available memory
is $_snmp_oid_val bytes"
  action 2.0 force-switchover

```

SNMP Notification Event Detector

The following example shows how to configure the **snmp-server community public RW** and **snmp-server manager** commands before **event snmp-notification** is configured.

```

snmp-server community public RW
snmp-server manager

```

The following example shows how an EEM applet called `SNMP_Notification` is being registered to run an EEM script when the device receives an SNMP notification on destination IP address 192.168.1.1 for object ID 1 whose value equals 10.


```
event manager applet SNMP_Notification
  event snmp-notification dest_ip_address 192.168.1.1 oid 1 op eq oid-value 10
  action 1 policy eem_script
```

Syslog Event Detector

The following example shows how to specify an EEM applet to run when syslog identifies that Ethernet interface 1/0 is down. The applet sends a message about the interface to syslog.

```
event manager applet interface-down
  event syslog pattern \.*UPDOWN.*Ethernet1/0.*" occurs 4
  action 1.0 syslog msg "Ethernet interface 1/0 changed state 4 times"
```

Configuration Examples for Embedded Event Manager Applet

Example Identity Event Detector

The following example shows how a policy named “EventIdentity” is triggered every time the authentication on the Fast Ethernet interface 0 is success.

```
event manager applet EventIdentity
  event identity interface FastEthernet0 authc success
  action 1.0 syslog msg "Applet EventIdentity"
```

Example MAT Event Detector

The following example shows how a policy named “EventMat” is triggered every time a mac-address is learned in the mac-address-table.

```
event manager applet EventMat
  event mat interface FastEthernet0
  action 1.0 syslog msg "Applet EventMat"
```

Example Neighbor-Discovery Event Detector

The following example shows how a policy named “EventNeighbor” is triggered when a Cisco Discovery Protocol (CDP) cache entry changes.

```
event manager applet EventNeighbor
  event neighbor-discovery interface FastEthernet0 cdp all
  action 1.0 syslog msg "Applet EventNeighbor"
```

Embedded Event Manager Manual Policy Execution Examples

The following examples show how to use the none event detector to configure an EEM policy (applet or script) to be run manually.

Using the event manager run Command

This example shows how to run a policy manually using the **event manager run** command. The policy is registered using the **event none** command under applet configuration mode and then run from global configuration mode using the **event manager run** command.

```

event manager applet manual-policy
  event none
  action 1.0 syslog msg "Manual-policy triggered"
  end
!
event manager run manual-policy

```

Using the action policy Command

This example shows how to run a policy manually using the **action policy** command. The policy is registered using the **event none** command under applet configuration mode, and then the policy is executed using the **action policy** command in applet configuration mode.

```

event manager applet manual-policy
  event none
  action 1.0 syslog msg "Manual-policy triggered"
  exit
!
event manager applet manual-policy-two
  event none
  action 1.0 policy manual-policy
  end
!
event manager run manual-policy-two

```

Embedded Event Manager Watchdog System Monitor (Cisco IOS) Event Detector Configuration Example

The following example shows how to configure three EEM applets to demonstrate how the Cisco IOS watchdog system monitor (IOSWDSysMon) event detector works.

Watchdog System Monitor Sample1 Policy

The first policy triggers an applet when the average CPU usage for the process named IP Input is greater than or equal to 1 percent for 10 seconds:

```

event manager applet IOSWD_Sample1
  event ioswdsysmon sub1 cpu-proc taskname "IP Input" op ge val 1 period 10
  action 1.0 syslog msg "IOSWD_Sample1 Policy Triggered"

```

Watchdog System Monitor Sample2 Policy

The second policy triggers an applet when the total amount of memory used by the process named Net Input is greater than 100 kb:

```

event manager applet IOSWD_Sample2
  event ioswdsysmon sub1 mem-proc taskname "Net Input" op gt val 100 is-percent false
  action 1.0 syslog msg "IOSWD_Sample2 Policy Triggered"

```

Watchdog System Monitor Sample3 Policy

The third policy triggers an applet when the total amount of memory used by the process named IP RIB Update has increased by more than 50 percent over the sample period of 60 seconds:

```

event manager applet IOSWD_Sample3
 event ioswdsysmon sub1 mem-proc taskname "IP RIB Update" op gt val 50 is-percent true
 period 60
 action 1.0 syslog msg "IOSWD_Sample3 Policy Triggered"

```

The three policies are configured, and then repetitive large pings are made to the networking device from several workstations, causing the networking device to register some usage. This will trigger policies 1 and 2, and the console will display the following messages:

```

00:42:23: %HA_EM-6-LOG: IOSWD_Sample1: IOSWD_Sample1 Policy Triggered
00:42:47: %HA_EM-6-LOG: IOSWD_Sample2: IOSWD_Sample2 Policy Triggered

```

To view the policies that are registered, use the **show event manager policy registered** command:

```

Device# show event manager policy registered
No.  Class  Type  Event Type  Trap  Time Registered  Name
1    applet  system ioswdsysmon  Off   Fri Jul 23 02:27:28 2004  IOSWD_Sample1
  sub1 cpu_util {taskname {IP Input} op ge val 1 period 10.000 }
  action 1.0 syslog msg "IOSWD_Sample1 Policy Triggered"
2    applet  system ioswdsysmon  Off   Fri Jul 23 02:23:52 2004  IOSWD_Sample2
  sub1 mem_used {taskname {Net Input} op gt val 100 is_percent FALSE}
  action 1.0 syslog msg "IOSWD_Sample2 Policy Triggered"
3    applet  system ioswdsysmon  Off   Fri Jul 23 03:07:38 2004  IOSWD_Sample3
  sub1 mem_used {taskname {IP RIB Update} op gt val 50 is_percent TRUE period 60.000 }
  action 1.0 syslog msg "IOSWD_Sample3 Policy Triggered"

```

Configuration SNMP Library Extensions Examples

SNMP Get Operations Examples

The following example shows how to send a get request to the local host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact
community
 public
Device(config-applet)# action 1.3 info type snmp oid
 1.3.6.1.2.1.1.4.0 get-type next community
public

```

The following log message will be written to the SNMP event manager log:

```

1d03h:%HA_EM-6-LOG: lg: 1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgn: 1.3.6.1.2.1.1.5.0

```

The following example shows how to send a get request to a remote host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val

```

```

5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 get-type next community
public ipaddr
172.17.16.69
Device(config-applet)# action 1.3 info type snmp getid
1.3.6.1.2.1.1.1.0 community
public ipaddr
172.17.16.69

```

The following log message is written to the SNMP event manager log:

```

1d03h:%HA_EM-6-LOG: lg: 1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgn: 1.3.6.1.2.1.1.5.0

```

SNMP GetID Operations Examples

The following example shows how to send a getid request to the local host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp getid
community
public

```

The following log message is written to the SNMP event manager log:

```

1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_oid=1.3.6.1.2.1.1.5.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_value=jubjub.cisco.com
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_oid=1.3.6.1.2.1.1.6.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_value=
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysdescr_oid=1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_oid=1.3.6.1.2.1.1.2.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_value=products.222
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=1.3.6.1.2.1.1.3.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=10131676
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_oid=1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_value=YYY

```

The following example shows how to send a getid request to a remote host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp getid
1.3.6.1.2.1.1.1.0 community
public ipaddr
172.17.16.69

```

The following log message is written to the SNMP event manager log:

```

1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_oid=1.3.6.1.2.1.1.5.0

```

```

1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_value=jubjub.cisco.com
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_oid=1.3.6.1.2.1.1.6.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_value=
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysdescr_oid=1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_oid=1.3.6.1.2.1.1.2.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_value=products.222
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=1.3.6.1.2.1.1.3.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=10131676
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_oid=1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_value=YYY

```

Set Operations Examples

The following example shows how to perform a set operation on the local host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 set-type
integer
5 sysName.0 community
public

```

The following log message is written to the SNMP event manager log:

```

1d04h:%HA_EM-6-LOG: lset: 1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lset: XXX

```

The following example shows how to perform a set operation on a remote host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 set-type integer
5 sysName.0 community
public ipaddr
172.17.16.69

```

The following log message is written to the SNMP event manager log:

```

1d04h:%HA_EM-6-LOG: lset: 1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lset: XXX

```

Generating SNMP Notifications Examples

The following example shows how to configure SNMP traps for the sysUpTime.0 variable:

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
lt entry-val

```

```

5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp var
sysUpTime.0 oid
1.3.6.1.4.1.9.9.43.1.1.6.1.3.41 integer
2
Device(config-applet)# action 1.4 info type snmp trap
enterprise-oid
ciscoSyslogMIB.2 generic-trapnum
6 specific-trapnum
1 trap-oid
1.3.6.1.4.1.9.9.41.2.0.1 trap-var
sysUpTime.0

```

The following output is generated if the debug snmp packets command is enabled:

```

Device# debug snmp packets
1d04h: SNMP: Queuing packet to 172.69.16.2
1d04h: SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 172.19.rap 1
clogHistoryEntry.3 = 4
clogHistoryEntry.6 = 9999
1d04h: SNMP: Queuing packet to 172.19.208.130
1d04h: SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 172.19.rap 1
clogHistoryEntry.3 = 4
clogHistoryEntry.6 = 9999
1d04h: SNMP: Packet sent via UDP to 172.69.16.2
1d04h: SNMP: Packet sent via UDP to 172.69.16.2
infra-view10:
Packet Dump:
30 53 02 01 00 04 04 63 6f 6d 6d a4 48 06 09 2b
06 01 04 01 09 09 29 02 40 04 ac 13 d1 17 02 01
06 02 01 01 43 04 00 9b 82 5d 30 29 30 12 06 0d
2b 06 01 04 01 09 09 29 01 02 03 01 03 02 01 04
30 13 06 0d 2b 06 01 04 01 09 09 29 01 02 03 01
06 02 02 27 0f
Received SNMPv1 Trap:
Community: comm
Enterprise: ciscoSyslogMIBNotificationPrefix
Agent-addr: 172.19.209.23
Enterprise Specific trap.
Enterprise Specific trap: 1
Time Ticks: 10191453
clogHistSeverity = error(4)
clogHistTimestamp = 9999

```

The following example shows how to configure SNMP inform requests for the sysUpTime.0 variable:

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp var
sysUpTime.0 oid
1.3.6.1.4.1.9.9.43.1.1.6.1.3.41 integer
2
Device(config-applet)# action 1.4 info type snmp inform
trap-oid
1.3.6.1.4.1.9.9.43.2.0.1 trap-var
sysUpTime.0 community

```

```
public ipaddr
172.19.209.24
```

The following output is generated if the debug snmp packets command is enabled:

```
Device# debug snmp packets
1d04h: SNMP: Inform request, reqid 24, errstat 0, erridx 0
sysUpTime.0 = 10244391
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.40 = 1
1d04h: SNMP: Packet sent via UDP to 172.19.209.24.162
1d04h: SNMP: Packet received via UDP from 172.19.209.24 on FastEthernet0/0
1d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
1d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
1d04h: SNMP: Inform request, reqid 25, errstat 0, erridx 0
sysUpTime.0 = 10244396
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.41 = 2
1d04h: SNMP: Packet sent via UDP to 172.19.209.24.162
1d04h: SNMP: Packet received via UDP from 172.19.209.24 on FastEthernet0/0
1d04h: SNMP: Response, reqid 25, errstat 0, erridx 0
1d04h: SNMP: Response, reqid 25, errstat 0, erridx 0
Device# debug snmp packets
5d04h: SNMP: Packet received via UDP from 172.19.209.23 on FastEthernet0/0
5d04h: SNMP: Inform request, reqid 24, errstat 0, erridx 0
sysUpTime.0 = 10244391
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.40 = 1
5d04h: dest if_index = 1
5d04h: dest ip addr= 172.19.209.24
5d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
5d04h: SNMP: Packet sent via UDP to 172.19.209.23.57748
5d04h: SNMP: Packet received via UDP from 172.19.209.23 on FastEthernet0/0
5d04h: SNMP: Inform request, reqid 25, errstat 0, erridx 0
```

Configuring Variable Logic for EEM Applets Examples

The following sections provide examples on some selected action commands. For information on all the action commands supporting variable logic within applets, see the table below.

In this example, conditional loops **while**, **if** and **foreach** are used to print data. Other action commands such as **action divide**, **action increment** and **action puts** are used to define the actions to be performed when the conditions are met.

```
event manager applet printdata
event none
action 100 set colors "red green blue"
action 101 set shapes "square triangle rectangle"
action 102 set i "1"
action 103 while $i lt 6
action 104   divide $i 2
action 105   if $_remainder eq 1
action 106     foreach_iterator "$colors"
action 107       puts newline "$_iterator "
action 108     end
action 109     puts ""
action 110   else
action 111     foreach_iterator "$shapes"
action 112       puts newline "$_iterator "
action 113     end
action 114     puts ""
```

```

action 115 end
action 116 increment i
action 117 end

```

When the event manager applet ex is run, the following output is obtained:

```

event manager run printdata
red green blue
square triangle rectange
red green blue
square triangle rectange
red green blue

```

In this example, two environment variables `poll_interface` and `max_rx_rate` are set to `F0/0` and `3` respectively. Every 30 seconds there is a poll on an interface for rx rate. If the rx rate is greater than the threshold, a syslog message is displayed.

This applet makes use of the `foreach` conditional statement to poll the interface, the `if` conditional block to compare the value under `RXPS` with `max_rx_rate` that was set in the EEM environment variable.

```

event manager environment poll_interfaces F0/0
event manager environment max_rx_rate 3
ev man app check_rx_rate
ev timer watchdog name rx_timer time 30
action 100 foreach int $poll_interfaces
action 101 cli command "en"
action 102 cli command "show int $int summ | beg ----"
action 103 foreach line $_cli_result "\n"
action 105 regexp ".*[0-9]+\s+[0-9]+\s+[0-9]+\s+[0-9]+\s+[0-9]+\s+([0-9+])\s+.*" $line
junk rxps
action 106 if $_regexp_result eq 1
action 107 if $rxps gt $max_rx_rate
action 108 syslog msg "Warning rx rate for $int is > than threshold. Current value is
$rxps
(threshold is $max_rx_rate)"
action 109 end
action 110 end
action 111 end
action 112 end

```

Example syslog message:

```

Oct 16 09:29:26.153: %HA_EM-6-LOG: c: Warning rx rate for F0/0 is > than threshold.
Current value is 4 (threshold is 3)
The output of show int F0/0 summ is of the format:

```

```
#show int f0/0 summ
```

```

*: interface is up
IHQ: pkts in input hold queue      IQD: pkts dropped from input queue
OHQ: pkts in output hold queue     OQD: pkts dropped from output queue
RXBS: rx rate (bits/sec)           RXPS: rx rate (pkts/sec)
TXBS: tx rate (bits/sec)           TXPS: tx rate (pkts/sec)
TRTL: throttle count

```

Interface	IHQ	IQD	OHQ	OQD	RXBS	RXPS	TXBS	TXPS	TRTL
* FastEthernet0/0	0	87283	0	0	0	0	0	0	0



Note To use other action commands supporting variable logic within applets, use the commands listed in the table below.

Table 184: Available action commands

Action Commands	Purpose
action add	Adds the value of two variables when an EEM applet is triggered.
action append	Appends the given value to the current value of a variable.
action break	Causes an immediate exit from a loop of actions within an EEM applet.
action comment	Adds comments to an applet when an EEM applet is triggered.
action context retrieve	Retrieves variables identified by a given set of context names.
action context save	Saves information across multiple policy triggers.
action continue	Continues with a loop of actions when an EEM applet is triggered.
action decrement	Decrements the value of a variable when an EEM applet is triggered.
action divide	Divides the dividend value by the given divisor value.
action else	Specifies the beginning of else conditional action when an EEM applet is triggered.
action elseif	Identifies the beginning of the else conditional action when an EEM applet is triggered.
action end	Specifies the identification of the end of an conditional action when an EEM applet is triggered.
action exit	Specifies an immediate exit from the running applet.
action foreach	Specifies the iteration of an input string using the given loop.
action gets	Gets an input from the local TTY in a synchronous manner when an EEM applet is triggered.
action if	Specifies the identification of the beginning of an conditional action when an EEM applet is triggered.
action if goto	Instructs the applet to jump to a given label if the conditional action is triggered.
action increment	Increments the value of a variable when an EEM applet is triggered.
action info type interface-names	Specifies the action of obtaining interface names when an EEM applet is triggered.
action info type snmp getid	Retrieves the individual variables from a Simple Network Management Protocol (SNMP) operation.
action info type snmp inform	Sends an SNMP inform requests when an EEM applet is triggered.

Action Commands	Purpose
action info type snmp oid	Specifies the type of SNMP get operation and the object that is triggered.
action info type snmp trap	Sends SNMP trap requests when an EEM applet is triggered.
action info type snmp var	Creates a variable for an SNMP object identifier (OID).
action multiply	Specifies the action of multiplying the variable value with another value.
action puts	Enables the action of printing data directly to the local console.
action regexp	Specifies the action of matching a regular expression.
action set (EEM)	Specifies the action of setting the value of a variable.
action string compare	Specifies the action of comparing two unequal strings.
action string equal	Specifies the action of verifying whether or not two strings are equal.
action string first	Specifies the action of returning the index on the first occurrence of a character.
action string index	Specifies the action of returning the characters specified by the index.
action string last	Specifies the action of returning the index on the last occurrence of a character.
action string length	Specifies the action of returning the number of characters in a string.
action string match	Specifies the action of returning 1 to the <code>\$_string_result</code> variable if the string matches the regular expression.
action string range	Specifies the action of storing a range of characters in a variable.
action string replace	Specifies the action of storing a new string by replacing the old string when an EEM applet is triggered.
action string tolower	Specifies the action of storing specific range of characters in a variable in lowercase.
action string toupper	Specifies the action of storing specific range of characters in a variable in uppercase.
action string trim	Specifies the action to trim a string when an EEM applet is triggered.
action string trimleft	Specifies the action to trim the characters of one string from the left.
action string trimright	Specifies the action to trim the characters one string from the right.
action subtract	Subtracts the value of a variable from another value.
action while	Specifies the action of identifying the beginning of a loop.

Configuring Event SNMP-Object Examples

The following example shows the SET operation and the value to set is in `$_snmp_value` and it is managed by the script. The example below saves the oid and its value as contexts to be retrieved later.

```

event manager applet snmp-object1
  description "APPLET SNMP-OBJ-1"
  event snmp-object oid 1.3.6.1.2.1.31.1.1.1.18 type string sync no skip no istable yes
  default 0
  action 1 syslog msg "SNMP-OBJ1:TRIGGERED" facility "SNMP_OBJ"
  action 2 context save key myoid variable "_snmp_oid"
  action 3 context save key myvalue variable "_snmp_value"

```

Configuring Description of an EEM Applet Examples

The following example shows how to add or modify the description for an Embedded Event Manager (EEM) applet that is run by sampling Simple Network Management Protocol (SNMP):

```

event manager applet test
  description "This applet looks for the word count in syslog messages"
  event syslog pattern "count"
  action 1 syslog msg hi

```

Additional References

The following sections provide references related to writing EEM policies Using the Cisco IOS CLI.

Related Documents

Related Topic	Document Title
EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples	Cisco IOS Embedded Event Manager Command Reference
Embedded Event Manager overview	Embedded Event Manager Overview module
Embedded Event Manager policy writing using Tcl	Writing Embedded Event Manager Policies Using Tcl module
Configuring enhanced object tracking	Configuring Enhanced Object Tracking module

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	--

MIBs

MIB	MIBs Link
CISCO-EMBEDDED-EVENT-MGR-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	--

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 185: Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI

Feature Name	Releases	Feature Information
Embedded Event Manager 4.0	15.2(5)E1	This feature was introduced and is supported only on c2960cx platform.



CHAPTER 89

Writing Embedded Event Manager Policies Using Tcl

This module describes how software developers can write and customize Embedded Event Manager (EEM) policies using Tool command language (Tcl) scripts to handle Cisco software faults and events. EEM is a policy-driven process by means of which faults in the Cisco software system are reported through a defined application programming interface (API). The EEM policy engine receives notifications when faults and other events occur. EEM policies implement recovery on the basis of the current state of the system and the actions specified in the policy for a given event. Recovery actions are triggered when the policy is run.

- [Prerequisites for Writing Embedded Event Manager Policies Using Tcl, on page 1829](#)
- [Information About Writing Embedded Event Manager Policies Using Tcl, on page 1830](#)
- [How to Write Embedded Event Manager Policies Using Tcl, on page 1836](#)
- [Configuration Examples for Writing Embedded Event Manager Policies Using Tcl, on page 1865](#)
- [Additional References, on page 1886](#)
- [Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI, on page 1887](#)

Prerequisites for Writing Embedded Event Manager Policies Using Tcl

- Before writing EEM policies, you should be familiar with the “ Embedded Event Manager Overview ” module.
- If you want to write EEM policies using the command-line interface (CLI) commands, you should be familiar with the “ Writing Embedded Event Manager Policies Using the Cisco IOS CLI ” module.

Information About Writing Embedded Event Manager Policies Using Tcl

EEM Policies

EEM offers the ability to monitor events and take informational or corrective action when the monitored events occur or reach a threshold. An EEM policy is an entity that defines an event and the actions to be taken when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the command-line interface (CLI) configuration. A script is a form of policy that is written in Tool Command Language (Tcl).

EEM Applet

An EEM applet is a concise method for defining event screening criteria and the actions to be taken when that event occurs. In EEM applet configuration mode, three types of configuration statements are supported. The event commands are used to specify the event criteria to trigger the applet to run, the action commands are used to specify an action to perform when the EEM applet is triggered, and the **set** command is used to set the value of an EEM applet variable. Currently only the `_exit_status` variable is supported for the **set** command.

Only one event configuration command is allowed within an applet configuration. When applet configuration submode is exited and no event command is present, a warning is displayed stating that no event is associated with the applet. If no event is specified, the applet is not considered registered. When no action is associated with the applet, events are still triggered but no actions are performed. Multiple action configuration commands are allowed within an applet configuration. Use the **show event manager policy registered** command to display a list of registered applets.

Before modifying an EEM applet, be aware that the existing applet is not replaced until you exit applet configuration mode. While you are in applet configuration mode modifying the applet, the existing applet may be executing. It is safe to modify the applet without unregistering it, because changes are written to a temporary file. When you exit applet configuration mode, the old applet is unregistered and the new version is registered.

Action configuration commands within an applet are uniquely identified using the *label* argument, which can be any string value. Actions are sorted within an applet in ascending alphanumeric key sequence using the *label* argument as the sort key, and they are run using this sequence. The same *label* argument can be used in different applets; the labels must be unique only within one applet.

The Embedded Event Manager schedules and runs policies on the basis of an event specification that is contained within the policy itself. When applet configuration mode is exited, EEM examines the event and action commands that are entered and registers the applet to be run when a specified event occurs.

For more details about writing EEM policies using the Cisco IOS CLI, see the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.

EEM Script

All Embedded Event Manager scripts are written in Tcl. Tcl is a string-based command language that is interpreted at run time. The version of Tcl supported is Tcl version 8.3.4 plus added script support. Scripts are defined using an ASCII editor on another device, not on the networking device. The script is then copied

to the networking device and registered with EEM. Tcl scripts are supported by EEM. As an enforced rule, Embedded Event Manager policies are short-lived run time routines that must be interpreted and executed in less than 20 seconds of elapsed time. If more than 20 seconds of elapsed time are required, the `maxrun` parameter may be specified in the `event_register` statement to specify any desired value.

EEM policies use the full range of the Tcl language’s capabilities. However, Cisco provides enhancements to the Tcl language in the form of Tcl command extensions that facilitate the writing of EEM policies. The main categories of Tcl command extensions identify the detected event, the subsequent action, utility information, counter values, and system information.

EEM allows you to write and implement your own policies using Tcl. Writing an EEM script involves:

- Selecting the event Tcl command extension that establishes the criteria used to determine when the policy is run.
- Defining the event detector options associated with detecting the event.
- Choosing the actions to implement recovery or respond to the detected event.

EEM Policy Tcl Command Extension Categories

There are different categories of EEM policy Tcl command extensions.



Note The Tcl command extensions available in each of these categories for use in all EEM policies are described in later sections in this document.

Table 186: EEM Policy Tcl Command Extension Categories

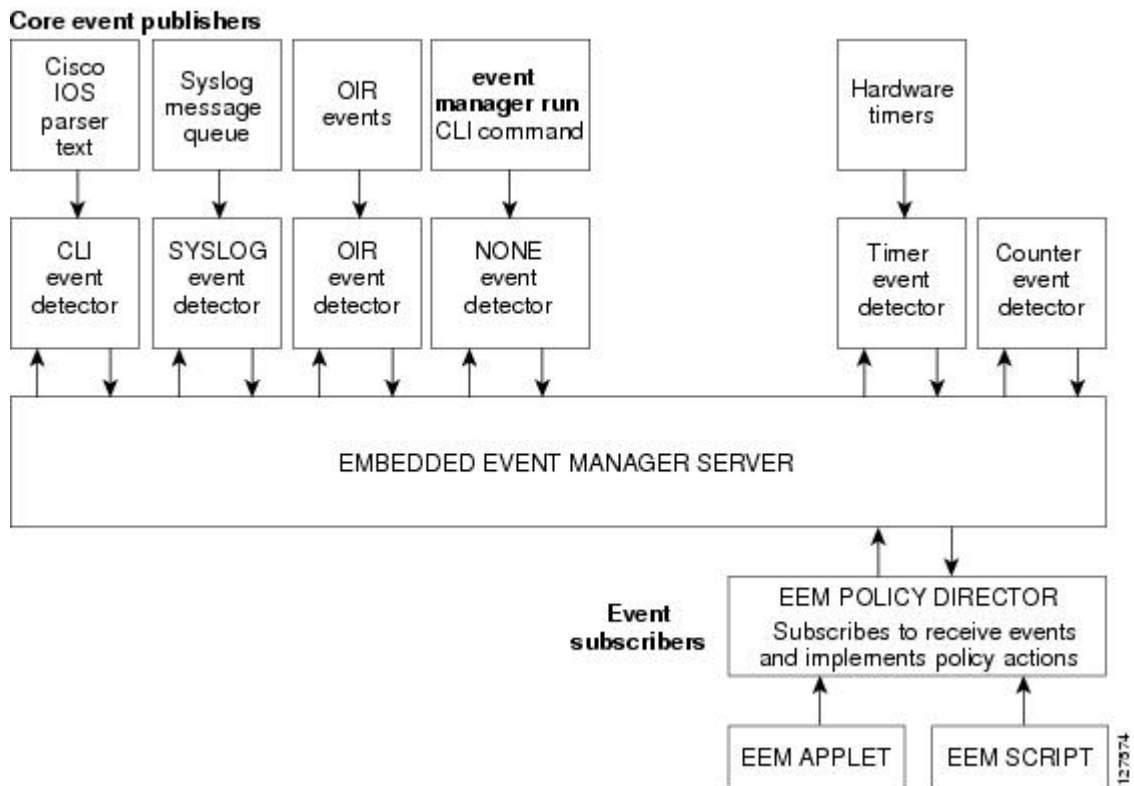
Category	Definition
EEM event Tcl command extensions (three types: event information, event registration, and event publish)	This category is represented by the event_register_ xxx family of event-specific commands. There is a separate event information Tcl command extension in this category as well: event_reqinfo . This is the command used in policies to query the EEM for information about an event. There is also an EEM event publish Tcl command extension event_publish > that publishes an application-specific event.
EEM action Tcl command extensions	These Tcl command extensions (for example, action_syslog) are used by policies to respond to or recover from an event or fault. In addition to these extensions, developers can use the Tcl language to implement any action desired.
EEM utility Tcl command extensions	These Tcl command extensions are used to retrieve, save, set, or modify application information, counters, or timers.
EEM system information Tcl command extensions	This category is represented by the sys_reqinfo_ xxx family of system-specific information commands. These commands are used by a policy to gather system information.
EEM context Tcl command extensions	These Tcl command extensions are used to store and retrieve a Tcl context (the visible variables and their values).

General Flow of EEM Event Detection and Recovery

EEM is a flexible, policy-driven framework that supports in-box monitoring of different components of the system with the help of software agents known as event detectors. The figure below shows the relationship between the EEM server, the core event publishers (event detectors), and the event subscribers (policies). Basically, event publishers screen events and publish them when there is a match on an event specification that is provided by the event subscriber. Event detectors notify the EEM server when an event of interest occurs.

When an event or fault is detected, Embedded Event Manager determines from the event publishers--an example would be the OIR events publisher in the figure below--if a registration for the encountered fault or event has occurred. EEM matches the event registration information with the event data itself. A policy registers for the detected event with the Tcl command extension `event_register _xxx`. The event information Tcl command extension `event_reqinfo` is used in the policy to query the Embedded Event Manager for information about the detected event.

Figure 133: Embedded Event Manager Core Event Detectors



Safe-Tcl

Safe-Tcl is a safety mechanism that allows untrusted Tcl scripts to run in an interpreter that was created in the safe mode. The safe interpreter has a restricted set of commands that prevent accessing some system resources and harming the host and other applications. For example, it does not allow commands to access critical Cisco IOS file system directories.

Cisco-defined scripts run in full Tcl mode, but user-defined scripts run in Safe-Tcl mode. Safe-Tcl allows Cisco to disable or customize individual Tcl commands. For more details about Tcl commands, go to <http://www.tcl.tk/man/>.

The following list of Tcl commands are restricted with a few exceptions. Restrictions are noted against each command or command keyword:

- **cd** --Change directory is not allowed to one of the restricted Cisco directory names.
- **encoding** --The commands **encoding names**, **encoding convertfrom**, and **encoding convertto** are permitted. The **encoding system** command with no arguments is permitted, but the **encoding system** command with the **?encoding?** keyword is not permitted.
- **exec** --Not permitted.
- **fconfigure** --Permitted.
- **file** --The following are permitted:
 - **file dirname**
 - **file exists**
 - **file extension**
 - **file isdirectory**
 - **file join**
 - **file pathtype**
 - **file rootname**
 - **file split**
 - **file stat**
 - **file tail**
- **file** --The following are not permitted:
 - **file atime**
 - **file attributes**
 - **file channels**
 - **file copy**
 - **file delete**
 - **file executable**
 - **file isfile**
 - **file link**
 - **file lstat**
 - **file mkdir**
 - **file mtime**
 - **file nativename**
 - **file normalize**
 - **file owned**
 - **file readable**
 - **file readlink**
 - **file rename**
 - **file rootname**
 - **file separator**
 - **file size**

- **file system**
 - **file type**
 - **file volumes**
 - **file writable**
- **glob** --The **glob** command is not permitted when searching in one of the restricted Cisco directories. Otherwise, it is permitted.
 - **load** --Only files that are in the user policy directory or the user library directory are permitted to be loaded. Static packages (for example, libraries that consist of C code) are not permitted to be loaded with the **load** command.
 - **open** --The **open** command is not allowed for a file that is located in one of the restricted Cisco directories.
 - **pwd** --The **pwd** command is not permitted.
 - **socket** --The **socket** command is permitted.
 - **source** --The **source** command is permitted for files that are in the user policy directory or the user library directory.

Bytecode Support for EEM 2.4

EEM 2.4 introduces bytecode language (BCL) support by accepting files with the standard bytecode script extension `.tbc`. Tcl version 8.3.4 defines a BCL and includes a compiler that translates Tcl scripts into BCL. Valid EEM policy file extensions in EEM 2.4 for user and system policies are `.tcl` (Tcl Text files) and `.tbc` (Tcl bytecode files).

Storing Tcl scripts in bytecode improves the execution speed of the policy because the code is precompiled, creates a smaller policy size, and obscures the policy code. Obfuscation makes it a little more difficult to modify scripts and hides logic to preserve intellectual property rights.

Support for bytecode is being added to provide another option for release of supported and trusted code. We recommend that you only run well understood, or trusted and supported software on network devices. To generate Tcl bytecode for IOS EEM support, use TclPro versions 1.4 or 1.5.

To translate a Tcl script to bytecode you can use `procomp`, part of Free TclPro Compiler, or Active State Tcl Development Kit. When a Tcl script is compiled using `procomp`, the code is scrambled and a `.tbc` file is generated. The bytecode files are platform-independent and can be generated on any operating system on which TclPro is available, including Windows, Linux, and UNIX. `Procomp` is part of TclPro and available from <http://www.tcl.tk/software/tclpro>.

Registration Substitution

In addition to regular Tcl substitution, EEM 2.3 permits the substitution of an individual parameter in an EEM event registration statement line with an environment variable.

EEM 2.4 introduces the ability to replace multiple parameters in event registration statement lines with a single environment variable.



Note Only the first environment variable supports multiple parameter substitution. Individual parameters can still be specified with additional environment variables after the initial variable.

To illustrate the substitution, a single environment variable, `$_eem_syslog_statement` is configured as:

```
::cisco::eem::event_register_syslog pattern COUNT
```

Using the registration substitution, the `$_eem_syslog_statement` environment variable is used in the following EEM user policy:

```
$_eem_syslog_statement occurs $_eem_occurs_val
action_syslog "this is test 3"
```

Environment variables must be defined before a policy using them is registered. To define the `$_eem_syslog_statement` environment variable:

```
Device(config)# event manager environment eem_syslog_statement
::cisco::eem::event_register_syslog pattern COUNT
Device(config)# event manager environment eem_occurs_val 2
```

Cisco File Naming Convention for EEM

All Embedded Event Manager policy names, policy support files (for example, e-mail template files), and library filenames are consistent with the Cisco file naming convention. In this regard, Embedded Event Manager policy filenames adhere to the following specification:

- An optional prefix--Mandatory.--indicating, if present, that this is a system policy that should be registered automatically at boot time if it is not already registered. For example: Mandatory.sl_text.tcl.
- A filename body part containing a two-character abbreviation (see the table below) for the first event specified; an underscore part; and a descriptive field part that further identifies the policy.
- A filename suffix part defined as .tcl.

Embedded Event Manager e-mail template files consist of a filename prefix of `email_template`, followed by an abbreviation that identifies the usage of the e-mail template.

Embedded Event Manager library filenames consist of a filename body part containing the descriptive field that identifies the usage of the library, followed by `_lib`, and a filename suffix part defined as .tcl.

Table 187: Two-Character Abbreviation Specification

ap	event_register_appl
cl	event_register_cli
ct	event_register_counter
go	event_register_gold
if	event_register_interface
io	event_register_ioswdsysmon
la	event_register_ipsla
nf	event_register_nf
no	event_register_none

oi	event_register_oir
pr	event_register_process
rf	event_register_rf
rs	event_register_resource
rt	event_register_routing
rp	event_register_rpc
sl	event_register_syslog
sn	event_register_snmp
st	event_register_snmp_notification
so	event_register_snmp_object
tm	event_register_timer
tr	event_register_track
ts	event_register_timer_subscriber
wd	event_register_wdysmon

How to Write Embedded Event Manager Policies Using Tcl

Registering and Defining an EEM Tcl Script

Perform this task to configure environment variables and register an EEM policy. EEM schedules and runs policies on the basis of an event specification that is contained within the policy itself. When an EEM policy is registered, the software examines the policy and registers it to be run when the specified event occurs.

Before you begin

You must have a policy available that is written in the Tcl scripting language. Sample policies are provided--see the details in the [Sample EEM Policies, on page 1847](#) to see which policies are available for the Cisco IOS release image that you are using--and these sample policies are stored in the system policy directory.

SUMMARY STEPS

1. **enable**
2. **show event manager environment** [**all**] *variable-name*
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. Repeat [Registering and Defining an EEM Tcl Script](#) to configure all the environment variables required by the policy to be registered in [Registering and Defining an EEM Tcl Script](#).

6. **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager environment [all <i>variable-name</i>] Example: Device# show event manager environment all	(Optional) Displays the name and value of EEM environment variables. <ul style="list-style-type: none"> • The optional all keyword displays all the EEM environment variables. • The optional <i>variable-name</i> argument displays information about the specified environment variable.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	event manager environment <i>variable-name string</i> Example: Device(config)# event manager environment _cron_entry 0-59/2 0-23/1 * * 0-6	Configures the value of the specified EEM environment variable. <ul style="list-style-type: none"> • In this example, the software assigns a CRON timer environment variable to be set to the second minute of every hour of every day.
Step 5	Repeat Registering and Defining an EEM Tcl Script to configure all the environment variables required by the policy to be registered in Registering and Defining an EEM Tcl Script .	--
Step 6	event manager policy <i>policy-filename</i> [type { system user }] [trap] Example: Device(config)# event manager policy tm_cli_cmd.tcl type system	Registers the EEM policy to be run when the specified event defined within the policy occurs. <ul style="list-style-type: none"> • Use the system keyword to register a Cisco-defined system policy. • Use the user keyword to register a user-defined system policy. • Use the trap keyword to generate an SNMP trap when the policy is triggered. • In this example, the sample EEM policy named <code>tm_cli_cmd.tcl</code> is registered as a system policy.

	Command or Action	Purpose
Step 7	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Examples

In the following example, the **show event manager environment** privileged EXEC command is used to display the name and value of all EEM environment variables.

```
Device# show event manager environment all
No.  Name                               Value
1    _cron_entry                          0-59/2 0-23/1 * * 0-6
2    _show_cmd                            show ver
3    _syslog_pattern                      .*UPDOWN.*Ethernet1/0.*
4    _config_cmd1                        interface Ethernet1/0
5    _config_cmd2                        no shut
```

Displaying EEM Registered Policies

Perform this optional task to display EEM registered policies.

SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*] [**time-ordered**|**name-ordered**] [**detailed** *policy-filename*]

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 show event manager policy registered [event-type *event-name*] [time-ordered|name-ordered] [detailed *policy-filename*]

Use this command with the **time-ordered** keyword to display information about currently registered policies sorted by time, for example:

Example:

```
Device# show event manager policy registered time-ordered
No.  Type      Event Type      Trap Time Registered      Name
1    system    timer cron        Off   Wed May11 01:43:18 2005  tm_cli_cmd.tcl
    name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
    nice 0 priority normal maxrun 240
```

```

2  system syslog                               Off  Wed May11  01:43:28 2005  sl_intf_down.tcl
   occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
   nice 0 priority normal maxrun 90
3  system proc abort                           Off  Wed May11  01:43:38 2005  pr_cdp_abort.tcl
   instance 1 path {cdp2.iosproc}
   nice 0 priority normal maxrun 20

```

Use this command with the **name-ordered** keyword to display information about currently registered policies sorted by name, for example:

Example:

```

Device# show event manager policy registered name-ordered
No.  Type      Event Type          Trap  Time Registered      Name
1    system  proc abort          Off   Wed May11  01:43:38 2005  pr_cdp_abort.tcl
   instance 1 path {cdp2.iosproc}
   nice 0 priority normal maxrun 20
2    system  syslog              Off   Wed May11  01:43:28 2005  sl_intf_down.tcl
   occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
   nice 0 priority normal maxrun 90
3    system  timer cron          Off   Wed May11  01:43:18 2005  tm_cli_cmd.tcl
   name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
   nice 0 priority normal maxrun 240

```

Use this command with the **event-type** keyword to display information about currently registered policies for the event type specified in the *event-name* argument, for example:

Example:

```

Device# show event manager policy registered event-type syslog
No.  Type      Event Type          Time Registered      Name
1    system  syslog              Wed May11  01:43:28 2005  sl_intf_down.tcl
   occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
   nice 0 priority normal maxrun 90

```

Unregistering EEM Policies

Perform this task to remove an EEM policy from the running configuration file. Execution of the policy is canceled.

SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [*event-type event-name*][*system| user*] [*time-ordered| name-ordered*] [*detailed policy-filename*]
3. **configure terminal**
4. **no event manager policy** *policy-filename*
5. **exit**
6. Repeat [Unregistering EEM Policies](#) to ensure that the policy has been removed.

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	show event manager policy registered [event-type <i>event-name</i>][system user] [time-ordered name-ordered] [detailed <i>policy-filename</i>] Example: Device# show event manager policy registered	(Optional) Displays the EEM policies that are currently registered. <ul style="list-style-type: none"> The optional system or user keyword displays the registered system or user policies. If no keywords are specified, EEM registered policies for all event types are displayed in time order.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	no event manager policy <i>policy-filename</i> Example: Device(config)# no event manager policy pr_cdp_terminate.tcl	Removes the EEM policy from the configuration, causing the policy to be unregistered. <ul style="list-style-type: none"> In this example, the no form of the command is used to unregister a specified policy.
Step 5	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 6	Repeat Unregistering EEM Policies to ensure that the policy has been removed. Example: Device# show event manager policy registered	--

Examples

In the following example, the **show event manager policy registered** privileged EXEC command is used to display the three EEM policies that are currently registered:

```
Device# show event manager policy registered
No.  Type   Event Type      Trap  Time Registered      Name
1    system timer cron          Off   Tue Oct11 01:43:18 2005 tm_cli_cmd.tcl
    name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
    nice 0 priority normal maxrun 240.000
2    system syslog          Off   Tue Oct11 01:43:28 2005 sl_intf_down.tcl
    occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
    nice 0 priority normal maxrun 90.000
3    system proc abort          Off   Tue Oct11 01:43:38 2005 pr_cdp_terminate.tcl
```



```
instance 1 path {cdp2.iosproc}
nice 0 priority normal maxrun 20.000
```

After the current policies are displayed, it is decided to delete the `pr_cdp_terminate.tcl` policy using the **no** form of the **event manager policy** command:

```
Device# configure terminal
Device(config)# no event manager policy pr_cdp_terminate.tcl
Device(config)# exit
```

The **show event manager policy registered** privileged EXEC command is entered again to display the EEM policies that are currently registered. The policy `pr_cdp_terminate.tcl` is no longer registered.

```
Device# show event manager policy registered
No.  Type      Event Type      Trap Time Registered      Name
1    system timer cron          Off  Tue Oct11 01:45:17 2005 tm_cli_cmd.tcl
   name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
   nice 0 priority normal maxrun 240.000
2    system syslog          Off  Tue Oct11 01:45:27 2005 sl_intf_down.tcl
   occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
   nice 0 priority normal maxrun 90.000
```

Suspending EEM Policy Execution

Perform this task to immediately suspend the execution of all EEM policies. Suspending policies, instead of unregistering them, might be necessary for reasons of temporary performance or security.

SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [*event-type event-name*][*system| user*] [*time-ordered| name-ordered*] [*detailed policy-filename*]
3. **configure terminal**
4. **event manager scheduler suspend**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>show event manager policy registered [<i>event-type event-name</i>][<i>system user</i>] [<i>time-ordered name-ordered</i>] [<i>detailed policy-filename</i>]</p> <p>Example:</p> <pre>Device# show event manager policy registered</pre>	<p>(Optional) Displays the EEM policies that are currently registered.</p> <ul style="list-style-type: none"> • The optional system or user keyword displays the registered system or user policies. • If no keywords are specified, EEM registered policies for all event types are displayed in time order.

	Command or Action	Purpose
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	event manager scheduler suspend Example: Device(config)# event manager scheduler suspend	Immediately suspends the execution of all EEM policies.
Step 5	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Examples

In the following example, the **show event manager policy registered** privileged EXEC command is used to display all the EEM registered policies:

```
Device# show event manager policy registered
No.  Type   Event Type      Trap  Time Registered      Name
1    system timer cron          Off   Sat Oct11 01:43:18 2003 tm_cli_cmd.tcl
    name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
    nice 0 priority normal maxrun 240.000
2    system syslog          Off   Sat Oct11 01:43:28 2003 sl_intf_down.tcl
    occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
    nice 0 priority normal maxrun 90.000
3    system proc abort      Off   Sat Oct11 01:43:38 2003 pr_cdp_abort.tcl
    instance 1 path {cdp2.iosproc}
    nice 0 priority normal maxrun 20.000
```

The **event manager scheduler suspend** command is entered to immediately suspend the execution of all EEM policies:

```
Device# configure terminal
Device(config)# event manager scheduler suspend
*Nov 2 15:34:39.000: %HA_EM-6-FMS_POLICY_EXEC: fh_io_msg: Policy execution has been
suspended
```

Managing EEM Policies

Perform this task to specify a directory to use for storing user library files or user-defined EEM policies.



Note This task applies only to EEM policies that are written using Tcl scripts.

SUMMARY STEPS

1. **enable**
2. **show event manager directory user [library| policy]**
3. **configure terminal**
4. **event manager directory user {library path| policy path}**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show event manager directory user [library policy] Example: <pre>Device# show event manager directory user library</pre>	(Optional) Displays the directory to use for storing EEM user library or policy files. <ul style="list-style-type: none"> • The optional library keyword displays the directory to use for user library files. • The optional policy keyword displays the directory to use for user-defined EEM policies.
Step 3	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 4	event manager directory user {library path policy path} Example: <pre>Device(config)# event manager directory user library disk0:/user_library Device(config)# event manager directory user library bootflash:/user_library</pre>	Specifies a directory to use for storing user library files or user-defined EEM policies. <ul style="list-style-type: none"> • Use the <i>path</i> argument to specify the absolute pathname to the user directory.
Step 5	exit Example: <pre>Device(config)# exit</pre>	Exits global configuration mode and returns to privileged EXEC mode.

Examples

In the following example, the **show event manager directory user** privileged EXEC command is used to display the directory, if it exists, to use for storing EEM user library files:

```
Device# show event manager directory user library
disk0:/user_library
```

```
Device# show event manager directory user library
bootflash:/user_library
```

Modifying History Table Size and Displaying EEM History Data

Perform this optional task to change the size of the history tables and to display EEM history data.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager history size {events | traps} [size]**
4. **exit**
5. **show event manager history events [detailed] [maximum number]**
6. **show event manager history traps [server | policy]**

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 configure terminal

Enters global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 event manager history size {events | traps} [size]

Use this command to change the size of the EEM event history table or the size of the EEM SNMP trap history table. In the following example, the size of the EEM event history table is changed to 30 entries:

Example:

```
Device(config)# event manager history size events 30
```

Step 4 exit

Exits global configuration mode and returns to privileged EXEC mode.

Example:

```
Device(config)# exit
```

Step 5 `show event manager history events [detailed] [maximum number]`

Use this command to display information about each EEM event that has been triggered.

Example:

```
Device# show event manager history events
No.  Time of Event          Event Type          Name
1    Fri Sep  9 13:48:40 2005  syslog             applet: one
2    Fri Sep  9 13:48:40 2005  syslog             applet: two
3    Fri Sep  9 13:48:40 2005  syslog             applet: three
4    Fri Sep  9 13:50:00 2005  timer cron         script: tm_cli_cmd.tcl
5    Fri Sep  9 13:51:00 2005  timer cron         script: tm_cli_cmd.tcl
```

Step 6 `show event manager history traps [server | policy]`

Use this command to display the EEM SNMP traps that have been sent either from the EEM server or from an EEM policy.

Example:

```
Device# show event manager history traps
No.  Time          Trap Type          Name
1    Fri Sep  9 13:48:40 2005  server            applet: four
2    Fri Sep  9 13:57:03 2005  policy            script: no_snmp_test.tcl
```

Displaying Software Modularity Process Reliability Metrics Using EEM

Perform this optional task to display reliability metrics for Cisco IOS Software Modularity processes. The `show event manager metric processes` command is supported only in Software Modularity images.

SUMMARY STEPS

1. `enable`
2. `show event manager metric process {all| process-name}`

DETAILED STEPS**Step 1** `enable`

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 `show event manager metric process {all| process-name}`

Use this command to display the reliability metric data for processes. The system keeps a record of when processes start and end, and this data is used as the basis for reliability analysis. In this partial example, the first and last entries showing the metric data for the processes on all the cards inserted in the system are displayed.

Example:

```

Device# show event manager metric process all
=====
process name: devc-pty, instance: 1
sub_system id: 0, version: 00.00.0000
-----
last event type: process start
recent start time: Fri Oct10 20:34:40 2005
recent normal end time: n/a
recent abnormal end time: n/a
number of times started: 1
number of times ended normally: 0
number of times ended abnormally: 0
most recent 10 process start times:
-----
Fri Oct10 20:34:40 2005
-----
most recent 10 process end times and types:
cumulative process available time: 6 hours 30 minutes 7 seconds 378 milliseconds
cumulative process unavailable time: 0 hours 0 minutes 0 seconds 0 milliseconds
process availability: 0.100000000
number of abnormal ends within the past 60 minutes (since reload): 0
number of abnormal ends within the past 24 hours (since reload): 0
number of abnormal ends within the past 30 days (since reload): 0
.
.
.
=====
process name: cdp2.iosproc, instance: 1
sub_system id: 0, version: 00.00.0000
-----
last event type: process start
recent start time: Fri Oct10 20:35:02 2005
recent normal end time: n/a
recent abnormal end time: n/a
number of times started: 1
number of times ended normally: 0
number of times ended abnormally: 0
most recent 10 process start times:
-----
Fri Oct10 20:35:02 2005
-----
most recent 10 process end times and types:

cumulative process available time: 6 hours 29 minutes 45 seconds 506 milliseconds
cumulative process unavailable time: 0 hours 0 minutes 0 seconds 0 milliseconds
process availability: 0.100000000
number of abnormal ends within the past 60 minutes (since reload): 0
number of abnormal ends within the past 24 hours (since reload): 0
number of abnormal ends within the past 30 days (since reload): 0

```

Troubleshooting Tips

Use the **debug event manager** command in privileged EXEC mode to troubleshoot EEM command operations. Use any debugging command with caution because the volume of output generated can slow or stop the device operations. We recommend that this command be used only under the supervision of a Cisco engineer.

Modifying the Sample EEM Policies

Perform this task to modify one of the sample policies. Cisco software contains some sample policies in the images that contain the Embedded Event Manager. Developers of EEM policies may modify these policies by customizing the event for which the policy is to be run and the options associated with logging and responding to the event. In addition, developers may select the actions to be implemented when the policy runs.

Sample EEM Policies

Cisco includes a set of sample policies shown in the table below. You can copy the sample policies to a user directory and then modify the policies, or you can write your own policies. Tcl is currently the only Cisco-supported scripting language for policy creation. Tcl policies can be modified using a text editor such as Emacs. Policies must execute within a defined number of seconds of elapsed time, and the time variable can be configured within a policy. The default is currently 20 seconds.

The table below describes the sample EEM policies.

Table 188: Sample EEM Policy Descriptions

Name of Policy	Description
pr_cdp_abort.tcl	Introduced with Cisco Software Modularity images. This policy monitors for cdp2.iosproc process termination events. It will log a message to SYSLOG and send an e-mail with the details of the termination.
pr_crash_reporter.tcl	Introduced with Cisco Software Modularity images. This policy monitors for all process termination events. When an event occurs, the policy will send crash information, including the crashdump file, to the specified URL where a CGI script processes the data.
pr_iprouting_abort.tcl	Introduced with Cisco Software Modularity images. This policy monitors for iprouting.iosproc process termination events. It will log a message to SYSLOG and send an e-mail with the details of the termination.
sl_intf_down.tcl	This policy runs when a configurable syslog message is logged. It will execute a configurable CLI command and e-mail the results.
tm_cli_cmd.tcl	This policy runs using a configurable CRON entry. It will execute a configurable CLI command and e-mail the results.
tm_crash_history.tcl	Introduced with Cisco Software Modularity images. This policy runs at midnight every day and e-mails a process crash history report to a specified e-mail address.
tm_crash_reporter.tcl	This policy runs 5 seconds after it is registered. If the policy is saved in the configuration, it will also run each time that the device is reloaded. The policy will prompt for the reload reason. If the reload was due to a crash, the policy will search for the latest crashinfo file and send this information to a specified URL location.
tm_fsyz_usage.tcl	Introduced with Cisco Software Modularity images. This policy runs using a configurable CRON entry and monitors disk space usage. A syslog message will be displayed if disk space usage crosses configurable thresholds.

Name of Policy	Description
wd_mem_reporter.tcl	Introduced with Cisco Software Modularity images. This policy reports on low system memory conditions when the amount of memory available falls below 20 percent of the initial available system memory. A syslog message will be displayed and, optionally, an e-mail will be sent.

For more details about the sample policies available and how to run them, see the [EEM Event Detector Demo Examples, on page 1865](#).

SUMMARY STEPS

1. **enable**
2. **show event manager policy available detailed** *policy-filename*
3. Cut and paste the contents of the sample policy displayed on the screen to a text editor.
4. Edit the policy and save it with a new filename.
5. Copy the new file back to the device flash memory.
6. **configure terminal**
7. **event manager directory user** {*library path*|*policy path*}
8. **event manager policy** *policy-filename* [**type** {*system*|*user*}] [**trap**]

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 show event manager policy available detailed *policy-filename*

Displays the actual specified sample policy including details about the environment variables used by the policy and instructions for running the policy. The **detailed** keyword was introduced for the **show event manager policy available** and the **show event manager policy registered** commands. Depending on your release, you may need to copy one of the two Tcl scripts from the configuration examples section in this document (see the [Programming Policies with Tcl Sample Scripts Example, on page 1873](#)). In the following example, details about the sample policy `tm_cli_cmd.tcl` are displayed on the screen.

Example:

```
Device# show event manager policy available detailed tm_cli_cmd.tcl
```

Step 3 Cut and paste the contents of the sample policy displayed on the screen to a text editor.

Use the edit and copy functions to move the contents from the device to a text editor on another device.

Step 4 Edit the policy and save it with a new filename.

Use the text editor to modify the policy as a Tcl script. For file naming conventions, see the [Cisco File Naming Convention for EEM, on page 1835](#).

Step 5 Copy the new file back to the device flash memory.

Copy the file to the flash file system on the device--typically disk0:. For more details about copying files, see the “Using the Cisco IOS File System” chapter in the *Configuration Fundamentals Configuration Guide*.

Copy the file to the flash file system on the device--typically bootflash:. For more details about copying files, see the “Using the Cisco IOS File System” chapter in the *Configuration Fundamentals Configuration Guide*.

Step 6 **configure terminal**

Enters global configuration mode.

Example:

```
Device# configure terminal
```

Step 7 **event manager directory user {library path} policy path}**

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the user_library directory on disk0 is specified as the directory for storing user library files.

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the user_library directory on bootflash is specified as the directory for storing user library files.

Example:

```
Device(config)# event manager directory user library disk0:/user_library
```

```
Device(config)# event manager directory user library bootflash:/user_library
```

Step 8 **event manager policy policy-filename [type {system| user}] [trap]**

Registers the EEM policy to be run when the specified event defined within the policy occurs. In the following example, the new EEM policy named test.tcl is registered as a user-defined policy.

Example:

```
Device(config)# event manager policy test.tcl type user
```

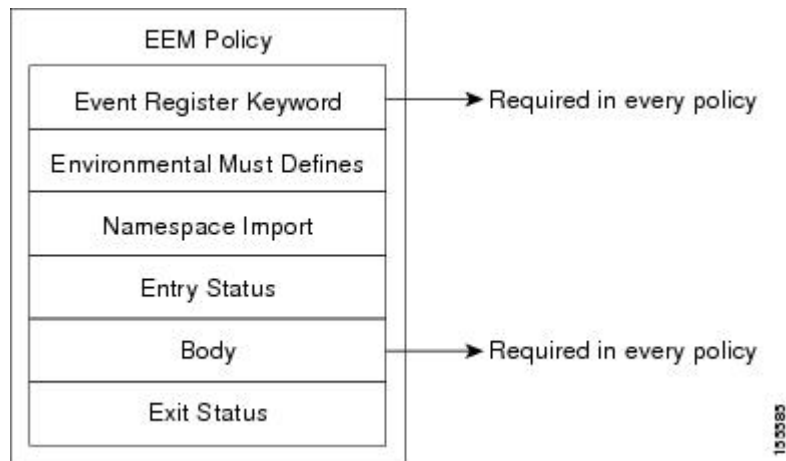
Programming EEM Policies with Tcl

Perform this task to help you program a policy using Tcl command extensions. We recommend that you copy an existing policy and modify it. There are two required parts that must exist in an EEM Tcl policy: the **event_register** Tcl command extension and the body. All other sections shown in the [Tcl Policy Structure and Requirements, on page 1849](#) concept are optional.

Tcl Policy Structure and Requirements

All EEM policies share the same structure, shown in the figure below. There are two parts of an EEM policy that are required: the **event_register** Tcl command extension and the body. The remaining parts of the policy are optional: environment must defines, namespace import, entry status, and exit status.

Figure 134: Tcl Policy Structure and Requirements



The start of every policy must describe and register the event to detect using an **event_register** Tcl command extension. This part of the policy schedules the running of the policy. The following example Tcl code shows how to register the **event_register_timer** Tcl command extension:

```
::cisco::eem::event_register_timer cron name crontimer2 cron_entry $_cron_entry maxrun 240
```

The environment must defines section is optional and includes the definition of environment variables. The following example Tcl code shows how to check for, and define, some environment variables.

```
# Check if all the env variables that we need exist.
# If any of them does not exist, print out an error msg and quit.
if {[info exists _email_server]} {
    set result \
        "Policy cannot be run: variable _email_server has not been set"
    error $result $errorInfo
}
if {[info exists _email_from]} {
    set result \
        "Policy cannot be run: variable _email_from has not been set"
    error $result $errorInfo
}
if {[info exists _email_to]} {
    set result \
        "Policy cannot be run: variable _email_to has not been set"
    error $result $errorInfo
}
```

The namespace import section is optional and defines code libraries. The following example Tcl code shows how to configure a namespace import section.

```
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

The body of the policy is a required structure and might contain the following:

- The **event_reqinfo** event information Tcl command extension that is used to query the EEM for information about the detected event.
- The action Tcl command extensions, such as **action_syslog**, that are used to specify EEM specific actions.

- The system information Tcl command extensions, such as **sys_reqinfo_routername**, that are used to obtain general system information.
- Use of the SMTP library (to send e-mail notifications) or the CLI library (to run CLI commands) from a policy.
- The **context_save** and **context_retrieve** Tcl command extensions that are used to save Tcl variables for use by other policies.

The following example Tcl code shows the code to query an event and log a message as part of the body section.

```
# Query the event info and log a message.
array set arr_einfo [event_reqinfo]

if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

global timer_type timer_time_sec
set timer_type $arr_einfo(timer_type)
set timer_time_sec $arr_einfo(timer_time_sec)

# Log a message.
set msg [format "timer event: timer type %s, time expired %s" \
    $timer_type [clock format $timer_time_sec]]

action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

EEM Entry Status

The entry status part of an EEM policy is used to determine if a prior policy has been run for the same event, and to determine the exit status of the prior policy. If the `_entry_status` variable is defined, a prior policy has already run for this event. The value of the `_entry_status` variable determines the return code of the prior policy.

Entry status designations may use one of three possible values: 0 (previous policy was successful), Not=0 (previous policy failed), and Undefined (no previous policy was executed).

EEM Exit Status

When a policy finishes running its code, an exit value is set. The exit value is used by the Embedded Event Manager to determine whether or not to apply the default action for this event, if any. A value of zero means do not perform the default action. A value of nonzero means perform the default action. The exit status will be passed to subsequent policies that are run for the same event.

EEM Policies and Cisco Error Number

Some EEM Tcl command extensions set a Cisco Error Number Tcl global variable `_cerrno`. Whenever `_cerrno` is set, four other Tcl global variables are derived from `_cerrno` and are set along with it (`_cerr_sub_num`, `_cerr_sub_err`, `_cerr_posix_err`, and `_cerr_str`).

For example, the **action_syslog** command in the example below sets these global variables as a side effect of the command execution:

```
action_syslog priority warning msg "A sample message generated by action_syslog"
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

`_cerrno`: 32-Bit Error Return Values

The `_cerrno` set by a command can be represented as a 32-bit integer of the following form:

XYSSSSSSSSSSSSSEEEEEEEEEPPPPPPPP

For example, the following error return value might be returned from an EEM Tcl command extension:

862439AE

This number is interpreted as the following 32-bit value:

10000110001001000011100110101110

This 32-bit integer is divided up into the five variables shown in the table below.

Table 189: `_cerrno`: 32-Bit Error Return Value Variables

Variable	Description
XY	The error class (indicates the severity of the error). This variable corresponds to the first two bits in the 32-bit error return value; 10 in the case above, which indicates CERR_CLASS_WARNING: See the table below for the four possible error class encodings specific to this variable.
SSSSSSSSSSSSSS	The subsystem number that generated the most recent error (13 bits = 8192 values). This is the next 13 bits of the 32-bit sequence, and its integer value is contained in <code>\$_cerr_sub_num</code> .
Variable	Description
EEEEEEEE	The subsystem specific error number (8 bits = 256 values). This segment is the next 8 bits of the 32-bit sequence, and the string corresponding to this error number is contained in <code>\$_cerr_sub_err</code> .
PPPPPPPP	The pass-through POSIX error code (9 bits = 512 values). This represents the last of the 32-bit sequence, and the string corresponding to this error code is contained in <code>\$_cerr_posix_err</code> .

Error Class Encodings for XY

The first variable, XY, references the possible error class encodings shown in the table below.

Table 190: Error Class Encodings

00	CERR_CLASS_SUCCESS
01	CERR_CLASS_INFO
10	CERR_CLASS_WARNING
11	CERR_CLASS_FATAL

An error return value of zero means SUCCESS.

SUMMARY STEPS

1. **enable**
2. **show event manager policy available detailed** *policy-filename*
3. Cut and paste the contents of the sample policy displayed on the screen to a text editor.
4. Define the required **event_register** Tcl command extension.
5. Add the appropriate namespace under the `::cisco` hierarchy.
6. Program the `must defines` section to check for each environment variable that is used in this policy.
7. Program the body of the script.
8. Check the entry status to determine if a policy has previously run for this event.
9. Check the exit status to determine whether or not to apply the default action for this event, if a default action exists.
10. Set Cisco Error Number (`_cerno`) Tcl global variables.
11. Save the Tcl script with a new filename, and copy the Tcl script to the device.
12. **configure terminal**
13. **event manager directory user** *{library path| policy path}*
14. **event manager policy** *policy-filename* [**type** *{system| user}*] [**trap**]
15. Cause the policy to execute, and observe the policy.
16. Use debugging techniques if the policy does not execute correctly.

DETAILED STEPS

Step 1

enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2

show event manager policy available detailed *policy-filename*

Displays the actual specified sample policy including details about the environment variables used by the policy and instructions for running the policy. The **detailed** keyword was introduced for the **show event manager policy available** and the **show event manager policy registered** commands. Depending on your release, you must copy one of the two Tcl scripts from the configuration examples section in this document (see the [Programming Policies with Tcl Sample](#)

[Scripts Example, on page 1873](#)). In the following example, details about the sample policy `tm_cli_cmd.tcl` are displayed on the screen.

Example:

```
Device# show event manager policy available detailed tm_cli_cmd.tcl
```

Step 3 Cut and paste the contents of the sample policy displayed on the screen to a text editor.

Use the edit and copy functions to move the contents from the device to a text editor on another device. Use the text editor to edit the policy as a Tcl script.

Step 4 Define the required **event_register** Tcl command extension.

Choose the appropriate **event_register** Tcl command extension from the table below for the event that you want to detect, and add it to the policy.

Table 191: EEM Event Registration Tcl Command Extensions

Event Registration Tcl Command Extensions
event_register_appl
event_register_cli
event_register_counter
event_register_gold
event_register_interface
event_register_ioswdsysmon
event_register_ipsla
event_register_nf
event_register_none
event_register_oir
event_register_process
event_register_resource
event_register_rf
event_register_routing
event_register_rpc
event_register_snmp
event_register_snmp_notification
event_register_snmp_object
event_register_syslog

Event Registration Tcl Command Extensions
event_register_timer
event_register_timer_subscriber
event_register_track
event_register_wdysysmon

Step 5 Add the appropriate namespace under the ::cisco hierarchy.

Policy developers can use the new namespace ::cisco in Tcl policies in order to group all the extensions used by Cisco IOS EEM. There are two namespaces under the ::cisco hierarchy, and the table below shows which category of EEM Tcl command extension belongs under each namespace.

Table 192: Cisco IOS EEM Namespace Groupings

Namespace	Category of Tcl Command Extension
::cisco::eem	EEM event registration
	EEM event information
	EEM event publish
	EEM action
	EEM utility
	EEM context library
	EEM system information
	CLI library
::cisco::lib	SMTP library

Note Make sure that you import the appropriate namespaces or use the qualified command names when using the above commands.

Step 6 Program the must defines section to check for each environment variable that is used in this policy.

This is an optional step. Must defines are a section of the policy that tests whether any EEM environment variables that are required by the policy are defined before the recovery actions are taken. The must defines section is not required if the policy does not use any EEM environment variables. EEM environment variables for EEM scripts are Tcl global variables that are defined external to the policy before the policy is run. To define an EEM environment variable, use the Embedded Event Manager configuration command **event manager environment** CLI command. By convention all Cisco EEM environment variables begin with “_” (an underscore). In order to avoid future conflict, customers are urged not to define new variables that start with “_”.

Note You can display the Embedded Event Manager environment variables set on your system by using the **show event manager environment** privileged EXEC command.

For example, Embedded Event Manager environment variables defined by the sample policies include e-mail variables. The sample policies that send e-mail must have the variables shown in the table below set in order to function properly.

The table below describes the e-mail-specific environment variables used in the sample EEM policies.

Table 193: E-mail-Specific Environmental Variables Used by the Sample Policies

Environment Variable	Description	Example
_email_server	A Simple Mail Transfer Protocol (SMTP) mail server used to send e-mail.	The e-mail server name can be in any one of the following template formats: <ul style="list-style-type: none"> • username:password@host • username@host • host
_email_to	The address to which e-mail is sent.	engineering@example.com
_email_from	The address from which e-mail is sent.	devtest@example.com
_email_cc	The address to which the e-mail must be copied.	manager@example.com

The following example of a must define section shows how to program a check for e-mail-specific environment variables.

Example of Must Defines

Example:

```

if {[info exists _email_server]} {
    set result \
        "Policy cannot be run: variable _email_server has not been set"
    error $result $errorInfo
}
if {[info exists _email_from]} {
    set result \
        "Policy cannot be run: variable _email_from has not been set"
    error $result $errorInfo
}
if {[info exists _email_to]} {
    set result \
        "Policy cannot be run: variable _email_to has not been set"
    error $result $errorInfo
}
if {[info exists _email_cc]} {
    set result \
        "Policy cannot be run: variable _email_cc has not been set"
    error $result $errorInfo
}

```

Step 7

Program the body of the script.

In this section of the script, you can define any of the following:

- The **event_reqinfo** event information Tcl command extension that is used to query the EEM for information about the detected event.
- The action Tcl command extensions, such as **action_syslog**, that are used to specify EEM specific actions.

- The system information Tcl command extensions, such as **sys_reqinfo_routername**, that are used to obtain general system information.
- The **context_save** and **context_retrieve** Tcl command extensions that are used to save Tcl variables for use by other policies.
- Use of the SMTP library (to send e-mail notifications) or the CLI library (to run CLI commands) from a policy.

Step 8 Check the entry status to determine if a policy has previously run for this event.

If the prior policy is successful, the current policy may or may not require execution. Entry status designations may use one of three possible values: 0 (previous policy was successful), Not=0 (previous policy failed), and Undefined (no previous policy was executed).

Step 9 Check the exit status to determine whether or not to apply the default action for this event, if a default action exists.

A value of zero means do not perform the default action. A value of nonzero means perform the default action. The exit status will be passed to subsequent policies that are run for the same event.

Step 10 Set Cisco Error Number (`_cerrno`) Tcl global variables.

Some EEM Tcl command extensions set a Cisco Error Number Tcl global variable `_cerrno`. Whenever `_cerrno` is set, four other Tcl global variables are derived from `_cerrno` and are set along with it (`_cerr_sub_num`, `_cerr_sub_err`, `_cerr_posix_err`, and `_cerr_str`).

For example, the **action_syslog** command in the example below sets these global variables as a side effect of the command execution:

Example:

```
action_syslog priority warning msg "A sample message generated by action_syslog
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

Step 11 Save the Tcl script with a new filename, and copy the Tcl script to the device.

Embedded Event Manager policy filenames adhere to the following specification:

- An optional prefix--Mandatory.--indicating, if present, that this is a system policy that should be registered automatically at boot time if it is not already registered. For example: Mandatory.sl_text.tcl.
- A filename body part containing a two-character abbreviation (see [EEM Policies and Cisco Error Number, on page 1852](#)) for the first event specified; an underscore character part; and a descriptive field part further identifying the policy.
- A filename suffix part defined as .tcl.

For more details, see the [Cisco File Naming Convention for EEM, on page 1835](#).

Copy the file to the flash file system on the device--typically disk0:. For more details about copying files, see the “Using the Cisco IOS File System” chapter in the Cisco IOS Configuration Fundamentals Configuration Guide .

Copy the file to the flash file system on the device--typically bootflash:. For more details about copying files, see the “Using the Cisco IOS File System” chapter in the Cisco IOS Configuration Fundamentals Configuration Guide .

Step 12 **configure terminal**

Enters global configuration mode.

Example:

```
Device# configure terminal
```

Step 13 **event manager directory user** {library path| policy path}

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the user_library directory on disk0 is specified as the directory for storing user library files.

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the user_library directory on bootflash is specified as the directory for storing user library files.

Example:

```
Device(config)# event manager directory user library disk0:/user_library
```

```
Device(config)# event manager directory user library bootflash:/user_library
```

Step 14 **event manager policy** policy-filename [type {system| user}] [trap]

Registers the EEM policy to be run when the specified event defined within the policy occurs. In the following example, the new EEM policy named cl_mytest.tcl is registered as a user-defined policy.

Example:

```
Device(config)# event manager policy cl_mytest.tcl type user
```

Step 15 Cause the policy to execute, and observe the policy.

To test that the policy runs, generate the conditions that will cause the policy to execute and observe that the policy runs as expected.

Step 16 Use debugging techniques if the policy does not execute correctly.

Use the Cisco IOS **debug event manager** CLI command with its various keywords to debug issues. Refer to the [Troubleshooting Tips, on page 1858](#) for details about using Tcl-specific keywords.

Troubleshooting Tips

- Use the **debug event manager tcl commands** CLI command to debug issues with Tcl extension commands. When enabled, this command displays all data that is passed in and read back from the TTY session that handles the CLI interactions. This data helps ensure users that the commands they are passing to the CLI are valid.
- The CLI library allows users to run CLI commands and obtain the output of commands in Tcl. Use the **debug event manager tcl cli-library** CLI command to debug issues with the CLI library.
- The SMTP library allows users to send e-mail messages to an SMTP e-mail server. Use the **debug event manager tcl smtp_library** CLI command to debug issues with the SMTP library. When enabled, this command displays all data that is passed in and read back from the SMTP library routines. This data helps ensure users that the commands they are passing to the SMTP library are valid.
- Tcl is a flexible language that allows you to override commands. For example, you can modify the **set** command and create a version of the **set** command that displays a message when a scalar variable is set.

When the **set** command is entered in a policy, a message is displayed anytime a scalar variable is set, and this provides a way to debug scalar variables. To view an example of this debugging technique, see the [Tracing Tcl set Command Operations Example, on page 1884](#).

To view examples of the some of these debugging techniques, see the [Debugging Embedded Event Manager Policies Examples, on page 1882](#).

Creating an EEM User Tcl Library Index

Perform this task to create an index file that contains a directory of all the procedures contained in a library of Tcl files. This task allows you to test library support in EEM Tcl. In this task, a library directory is created to contain the Tcl library files, the files are copied into the directory, and an index (tclIndex) is created that contains a directory of all the procedures in the library files. If the index is not created, the Tcl procedures will not be found when an EEM policy is run that references a Tcl procedure.

SUMMARY STEPS

1. On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl library files into the directory.
2. **telsh**
3. **auto_mkindex** *directory_name* *.tcl
4. Copy the Tcl library files and the tclIndex file to the directory used for storing user library files on the target device.
5. Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.
6. **enable**
7. **configure terminal**
8. **event manager directory user library** *path*
9. **event manager directory user policy** *path*
10. **event manager policy** *policy-name* [type {system | user}] [trap]
11. **event manager run** *policy-name*

DETAILED STEPS

Step 1

On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl library files into the directory.

The following example files can be used to create a tclIndex on a workstation running the Tcl shell:

lib1.tcl

Example:

```
proc test1 {} {  
    puts "In procedure test1"  
}  
  
proc test2 {} {  
    puts "In procedure test2"  
}
```

lib2.tcl**Example:**

```
proc test3 {} {
    puts "In procedure test3"
}
```

Step 2 **tclsh**

Use this command to enter the Tcl shell.

Example:

```
workstation% tclsh
```

Step 3 **auto_mkindex** *directory_name* *.tcl

Use the **auto_mkindex** command to create the tclIndex file. The tclIndex file that contains a directory of all the procedures contained in the Tcl library files. We recommend that you run auto_mkindex inside a directory because there can only be a single tclIndex file in any directory and you may have other Tcl files to be grouped together. Running auto_mkindex in a directory determines which tcl source file or files are indexed using a specific tclIndex.

Example:

```
workstation% auto_mkindex eem_library *.tcl
```

The following example TclIndex is created when the lib1.tcl and lib2.tcl files are in a library file directory and the **auto_mkindex** command is run.

tclIndex**Example:**

```
# Tcl autoload index file, version 2.0
# This file is generated by the "auto_mkindex" command
# and sourced to set up indexing information for one or
# more commands. Typically each line is a command that
# sets an element in the auto_index array, where the
# element name is the name of a command and the value is
# a script that loads the command.

set auto_index(test1) [list source [file join $dir lib1.tcl]]
set auto_index(test2) [list source [file join $dir lib1.tcl]]
set auto_index(test3) [list source [file join $dir lib2.tcl]]
```

Step 4 Copy the Tcl library files and the tclIndex file to the directory used for storing user library files on the target device.

Step 5 Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.

The directory for storing user-defined EEM policies can be the same directory used in Step 4. The following example user-defined EEM policy can be used to test the Tcl library support in EEM.

libtest.tcl**Example:**

```
::cisco::eem::event_register_none
```

```
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

```

global auto_index auto_path

puts [array names auto_index]

if { [catch {test1} result]} {
    puts "calling test1 failed result = $result $auto_path"
}

if { [catch {test2} result]} {
    puts "calling test2 failed result = $result $auto_path"
}

if { [catch {test3} result]} {
    puts "calling test3 failed result = $result $auto_path"
}

```

Step 6 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 7 **configure terminal**

Enables global configuration mode.

Example:

```
Device# configure terminal
```

Step 8 **event manager directory user library *path***

Use this command to specify the EEM user library directory; this is the directory to which the files were copied.

Example:

```
Device(config)# event manager directory user library disk2:/eem_library
```

Step 9 **event manager directory user policy *path***

Use this command to specify the EEM user policy directory; this is the directory to which the file was copied.

Example:

```
Device(config)# event manager directory user policy disk2:/eem_policies
```

Step 10 **event manager policy *policy-name* [type {system | user}] [trap]**

Use this command to register a user-defined EEM policy. In this example, the policy named libtest.tcl is registered.

Example:

```
Device(config)# event manager policy libtest.tcl
```

Step 11 **event manager run *policy-name***

Use this command to manually run an EEM policy. In this example, the policy named libtest.tcl is run to test the Tcl support in EEM. The example output shows that the test for Tcl support in EEM was successful.

Example:

```
Device(config)# event manager run libtest.tcl
The following output is displayed:
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test1
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test2
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test3
```

Creating an EEM User Tcl Package Index

Perform this task to create a Tcl package index file that contains a directory of all the Tcl packages and version information contained in a library of Tcl package files. Tcl packages are supported, depending on your release, using the Tcl **package** keyword.

Tcl packages are located in either the EEM system library directory or the EEM user library directory. When a **package require** Tcl command is executed, the user library directory is searched first for a pkgIndex.tcl file. If the pkgIndex.tcl file is not found in the user directory, the system library directory is searched. In this task, a Tcl package directory--the pkgIndex.tcl file--is created in the appropriate library directory using the **pkg_mkIndex** command to contain information about all of the Tcl packages contained in the directory along with version information. If the index is not created, the Tcl packages will not be found when an EEM policy is run that contains a **package require** Tcl command.

Using the Tcl package support in EEM, users can gain access to packages such as XML_RPC for Tcl. When the Tcl package index is created, a Tcl script can easily make an XML-RPC call to an external entity.



Note Packages implemented in C programming code are not supported in EEM.

SUMMARY STEPS

1. On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl package files into the directory.
2. **tclsh**
3. **pkg_mkindex** *directory_name* *.tcl
4. Copy the Tcl library files and the pkgIndex file to the directory used for storing user library files on the target device.
5. Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.
6. **enable**
7. **configure terminal**
8. **event manager directory user library** *path*
9. **event manager directory user policy** *path*
10. **event manager policy** *policy-name* [**type** {system | user}] [**trap**]
11. **event manager run** *policy-name*

DETAILED STEPS

Step 1 On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl package files into the directory.

Step 2 **tclsh**

Use this command to enter the Tcl shell.

Example:

```
workstation% tclsh
```

Step 3 **pkg_mkindex** *directory_name* *.tcl

Use the **pkg_mkindex** command to create the pkgIndex file. The pkgIndex file contains a directory of all the packages contained in the Tcl library files. We recommend that you run pkg_mkindex inside a directory because there can only be a single pkgIndex file in any directory and you may have other Tcl files to be grouped together. Running pkg_mkindex in a directory determines which Tcl package file or files are indexed using a specific pkgIndex.

Example:

```
workstation% pkg_mkindex eem_library *.tcl
```

The following example pkgIndex is created when some Tcl package files are in a library file directory and the **pkg_mkindex** command is run.

pkgIndex

Example:

```
# Tcl package index file, version 1.1
# This file is generated by the "pkg_mkIndex" command
# and sourced either when an application starts up or
# by a "package unknown" script. It invokes the
# "package ifneeded" command to set up package-related
# information so that packages will be loaded automatically
# in response to "package require" commands. When this
# script is sourced, the variable $dir must contain the
# full path name of this file's directory.
package ifneeded xmlrpc 0.3 [list source [file join $dir xmlrpc.tcl]]
```

Step 4 Copy the Tcl library files and the pkgIndex file to the directory used for storing user library files on the target device.

Step 5 Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.

The directory for storing user-defined EEM policies can be the same directory used in Step 4. The following example user-defined EEM policy can be used to test the Tcl package support in EEM.

packagetest.tcl

Example:

```
::cisco::eem::event_register_none maxrun 1000000.000
#
# test if xmlrpc available
#
#
# Namespace imports
```

```
#
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
#
package require xmlrpc
puts "Did you get an error?"
```

Step 6 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 7 **configure terminal**

Enables global configuration mode.

Example:

```
Device# configure terminal
```

Step 8 **event manager directory user library** *path*

Use this command to specify the EEM user library directory; this is the directory to which the files in were copied.

Example:

```
Device(config)# event manager directory user library disk2:/eem_library
```

Step 9 **event manager directory user policy** *path*

Use this command to specify the EEM user policy directory; this is the directory to which the file was copied.

Example:

```
Device(config)# event manager directory user policy disk2:/eem_policies
```

Step 10 **event manager policy** *policy-name* [**type** {system | user}] [**trap**]

Use this command to register a user-defined EEM policy. In this example, the policy named packetest.tcl is registered.

Example:

```
Device(config)# event manager policy packetest.tcl
```

Step 11 **event manager run** *policy-name*

Use this command to manually run an EEM policy. In this example, the policy named packetest.tcl is run to test the Tcl package support in EEM.

Example:

```
Device(config)# event manager run packetest.tcl
```


Configuration Examples for Writing Embedded Event Manager Policies Using Tcl

Assigning a Username for a Tcl Session Examples

The following example shows how to set a username to be associated with a Tcl session. If you are using authentication, authorization, and accounting (AAA) security and implement authorization on a command basis, you should use the **event manager session cli username** command to set a username to be associated with a Tcl session. The username is used when a Tcl policy executes a CLI command. TACACS+ verifies each CLI command using the username associated with the Tcl session that is running the policy. Commands from Tcl policies are not usually verified because the device must be in privileged EXEC mode to register the policy. In the example, the username is yourname, and this is the username that is used whenever a CLI command session is initiated from within an EEM policy.

```
configure terminal
event manager session cli username yourname
end
```

EEM Event Detector Demo Examples

EEM Sample Policy Descriptions

This configuration example features some of the sample EEM policies:

- `ap_perf_test_base_cpu.tcl`--Is run to measure the the CPU performance of EEM policies.
- `no_perf_test_init.tcl`--Is run to measure the CPU performance of EEM policies.
- `sl_intf_down.tcl`--Is run when a configurable syslog message is logged. It executes up to two configurable CLI commands and e-mails the results.
- `tm_cli_cmd.tcl`--Is run using a configurable CRON entry. It executes a configurable CLI command and e-mails the results.
- `tm_crash_reporter.tcl`--Is run 5 seconds after it is registered and 5 seconds after the device boots up. When triggered, the script attempts to find the reload reason. If the reload reason was due to a crash, the policy searches for the related crashinfo file and sends this information to a URL location specified by the user in the environment variable `_crash_reporter_url`.
- `tm_fsys_usage.tcl`--This policy runs using a configurable CRON entry and monitors disk space usage. A syslog message is displayed if disk space usage crosses configurable thresholds.

Event Manager Environment Variables for the Sample Policies

Event manager environment variables are Tcl global variables that are defined external to the EEM policy before the policy is registered and run. The sample policies require three of the e-mail environment variables to be set ; only `_email_cc` is optional. Other required and optional variable settings are outlined in the following tables.

The table below describes the EEM environment variables that must be set before the `ap_perf_test_base_cpu.tcl` sample policy is run.

Table 194: Environment Variables Used in the `ap_perf_test_base_cpu.tcl` Policy

Environment Variable	Description	Example
<code>_perf_iterations</code>	The number of iterations over which to run the measurement.	100
<code>_perf_cmd1</code>	The first non interactive CLI command that is executed as part of the measurement test. This variable is optional and need not be specified.	enable
<code>_perf_cmd2</code>	The second non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd2</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified.	show version
<code>_perf_cmd3</code>	The third non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd3</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified.	show interface counters protocol status

The table below describes the EEM environment variables that must be set before the `no_perf_test_init.tcl` sample policy is run.

Table 195: Environment Variables Used in the `no_perf_test_init.tcl` Policy

Environment Variable	Description	Example
<code>_perf_iterations</code>	The number of iterations over which to run the measurement.	100
<code>_perf_cmd1</code>	The first non interactive CLI command that is executed as part of the measurement test. This variable is optional and need not be specified.	enable
<code>_perf_cmd2</code>	The second non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd2</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified.	show version
<code>_perf_cmd3</code>	The third non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd3</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified.	show interface counters protocol status

The table below describes the EEM environment variables that must be set before the `sl_intf_down.tcl` sample policy is run.

Table 196: Environment Variables Used in the `sl_intf_down.tcl` Policy

Environment Variable	Description	Example
<code>_config_cmd1</code>	The first configuration command that is executed.	interface Ethernet1/0

Environment Variable	Description	Example
_config_cmd2	The second configuration command that is executed. This variable is optional and need not be specified.	no shutdown
_syslog_pattern	A regular expression pattern match string that is used to compare syslog messages to determine when the policy runs.	. *UPDOWN.*FastEthernet0/0.*

The table below describes the EEM environment variables that must be set before the tm_cli_cmd.tcl sample policy is run.

Table 197: Environment Variables Used in the tm_cli_cmd.tcl Policy

Environment Variable	Description	Example
_cron_entry	A CRON specification that determines when the policy will run.	0-59/1 0-23/1 * * 0-7
_show_cmd	The CLI command to be executed when the policy is run.	show version

The table below describes the EEM environment variables that must be set before the tm_crash_reporter.tcl sample policy is run.

Table 198: Environment Variables Used in the tm_crash_reporter.tcl Policy

Environment Variable	Description	Example
_crash_reporter_debug	A value that identifies whether debug information for tm_crash_reporter.tcl will be enabled. This variable is optional and need not be specified.	1
_crash_reporter_url	The URL location to which the crash report is sent.	http://www.example.com/fm/interface_tm.cgi

The table below describes the EEM environment variables that must be set before the tm_fsys_usage.tcl sample policy is run.

Table 199: Environment Variables Used in the tm_fsys_usage.tcl Policy

Environment Variable	Description	Example
_tm_fsys_usage_cron	A CRON specification that is used in the event_register Tcl command extension. If unspecified, the tm_fsys_usage.tcl policy is triggered once per minute. This variable is optional and need not be specified.	0-59/1 0-23/1 * * 0-7
_tm_fsys_usage_debug	When this variable is set to a value of 1, disk usage information is displayed for all entries in the system. This variable is optional and need not be specified.	1

Environment Variable	Description	Example
_tm_fsys_usage_freebytes	Free byte threshold for systems or specific prefixes. If free space falls below a given value, a warning is displayed. This variable is optional and need not be specified.	disk2:98000000
_tm_fsys_usage_percent	Disk usage percentage thresholds for systems or specific prefixes. If the disk usage percentage exceeds a given percentage, a warning is displayed. If unspecified, the default disk usage percentage is 80 percent for all systems. This variable is optional and need not be specified.	nvr:25 disk2:5

Registration of Some EEM Policies

Some EEM policies must be unregistered and then reregistered if an EEM environment variable is modified after the policy is registered. The `event_register_xxx` statement that appears at the start of the policy contains some of the EEM environment variables, and this statement is used to establish the conditions under which the policy is run. If the environment variables are modified after the policy has been registered, the conditions may become invalid. To avoid any errors, the policy must be unregistered and then reregistered. The following variables are affected:

- `_cron_entry` in the `tm_cli_cmd.tcl` policy
- `_syslog_pattern` in the `sl_intf_down.tcl` policy

Basic Configuration Details for All Sample Policies

To allow e-mail to be sent from the Embedded Event Manager, the `hostname` and `ip domain-name` commands must be configured. The EEM environment variables must also be set. After a Cisco IOS image has been booted, use the following initial configuration, substituting appropriate values for your network. The environment variables for the `tm_fsys_usage` sample policy (see the table above) are all optional and are not listed here:

```
hostname cpu
ip domain-name example.com
event manager environment _email_server ms.example.net
event manager environment _email_to username@example.net
event manager environment _email_from engineer@example.net
event manager environment _email_cc projectgroup@example.net
event manager environment _cron_entry 0-59/2 0-23/1 * * 0-7
event manager environment _show_cmd show event manager policy registered
event manager environment _syslog_pattern .*UPDOWN.*FastEthernet0/0
event manager environment _config_cmd1 interface Ethernet1/0
event manager environment _config_cmd2 no shutdown
event manager environment _crash_reporter_debug 1
event manager environment _crash_reporter_url
http://www.example.com/fm/interface_tm.cgi
end
```

Using the Sample Policies

This section contains the following configuration scenarios to demonstrate how to use the some sample Tcl policies:

Running the Mandatory.go_*.tcl Sample Policy

There are GOLD TCL scripts for each test which runs as a part of GOLD EEM Policy. You can modify the TCL script for the test, specify the consecutive failure count, and also change the default corrective action. For example, one could choose to power down a linecard card, instead of reset or other CLI based actions.

For each registered test, a default TCL script is available, which can be registered with the system, and matches with the default action. This can be then overridden by modifying these scripts.

The following table shows a list of the mandatory policies that GOLD installed into EEM. Each of the policies performs some sort of action such as resetting the card or disabling the port.

GOLD Tcl Scripts	Test
Mandatory.go_asicsync.tcl	TestAsicSync
Mandatory.go_bootup.tcl	Common for all bootup tests.
Mandatory.go_fabric.tcl	TestFabricHealth
Mandatory.go_fabrich0.tcl	TestFabricCh0Health
Mandatory.go_fabrich1.tcl	TestFabricCh1Health
Mandatory.go_ipsec.tcl	TestIPSecEncrypDecrypPkt
Mandatory.go_mac.tcl	TestMacNotification
Mandatory.go_nondislp.tcl	TestNonDisruptiveLoopback
Mandatory.go_scratchreg.tcl	TestScratchRegister
Mandatory.go_sprping.tcl	TestSPRPIbandPing

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the mandatory.go_*.tcl policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
  event manager policy Mandatory.go_spuriousisr.tcl
end
show event manager policy registered
show event manager environment
```

Running the ap_perf_test_base_cpu.tcl and no_perf_test_init.tcl Sample Policies

These sample policies measure the CPU performance of EEM policies. The policies help find the average execution time of each EEM policy and use the CLI library to execute the configuration commands specified in the EEM environment variables `_perf_cmd1` and, optionally, `_perf_cmd2` and `_perf_cmd3`.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, enter the **service timestamps debug datetime msec** command and then you can register the `ap_perf_test_base_cpu.tcl` and `no_perf_test_init.tcl` policies with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

The policies `ap_perf_test_base_cpu.tcl` and `no_perf_test_init.tcl` need to be registered together, as they run as a test suite. You can run the `no_perf_test_init.tcl` policy to start the tests. Analyze the results using the syslog messages from each iteration. The total number of iteration is specified by the variable `_perf_iterations`. Take the time difference and divide it by the total number of iterations to get the average execution time of each EEM policy.

```
enable
show event manager policy registered
show event manager policy available
show event manager environment
configure terminal
  service timestamps debug datetime msec
  event manager environment _perf_iterations 100
  event manager policy ap_perf_test_base_cpu.tcl
  event manager policy no_perf_test_init.tcl
end
show event manager policy registered
show event manager policy available
show event manager environment
event manager run no_perf_test_init.tcl
```

Running the `no_perf_test_init.tcl` Sample Policy

This sample policy measures the the cpu performance of EEM policies. The policy helps to find the average execution time of each EEM policy and uses the CLI library to execute the configuration commands specified in the EEM environment variables `_perf_cmd1` and, optionally, `_perf_cmd2` and `_perf_cmd3`.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `no_perf_test_init.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

Analyze the results using the syslog messages from each iteration. The total number of iteration is specified by the variable `_perf_iterations`. Take the time difference and divide it by the total number of iterations to get the average execution time of each EEM policy.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
  event manager policy no_perf_test_init.tcl
end
```

```
show event manager policy registered
show event manager environment
```

Running the `sl_intf_down.tcl` Sample Policy

This sample policy demonstrates the ability to modify the configuration when a syslog message with a specific pattern is logged. The policy gathers detailed information about the event and uses the CLI library to execute the configuration commands specified in the EEM environment variables `_config_cmd1` and, optionally, `_config_cmd2`. An e-mail message is sent with the results of the CLI command.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `sl_intf_down.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

The policy runs when an interface goes down. Enter the **show event manager environment** command to display the current environment variable values. Unplug the cable (or configure a shutdown) for the interface specified in the `_syslog_pattern` EEM environment variable. The interface goes down, prompting the syslog daemon to log a syslog message about the interface being down, and the syslog event detector is called.

The syslog event detector reviews the outstanding event specifications and finds a match for interface status change. The EEM server is notified, and the server runs the policy that is registered to handle this event--`sl_intf_down.tcl`.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
  event manager policy sl_intf_down.tcl
end
show event manager policy registered
show event manager environment
```

Running the `tm_cli_cmd.tcl` Sample Policy

This sample policy demonstrates the ability to periodically execute a CLI command and to e-mail the results. The CRON specification “0-59/2 0-23/1 * * 0-7” causes this policy to be run on the second minute of each hour. The policy gathers detailed information about the event and uses the CLI library to execute the configuration commands specified in the EEM environment variable `_show_cmd`. An e-mail message is sent with the results of the CLI command.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `tm_cli_cmd.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command to verify that the policy has been registered.

The timer event detector triggers an event for this case periodically according to the CRON string set in the EEM environment variable `_cron_entry`. The EEM server is notified, and the server runs the policy that is registered to handle this event--`tm_cli_cmd.tcl`.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
  event manager policy tm_cli_cmd.tcl
end
show event manager policy registered
```

Running the `tm_crash_reporter.tcl` Sample Policy

This sample policy demonstrates the ability to send an HTTP-formatted crash report to a URL location. If the policy registration is saved in the startup configuration file, the policy is triggered 5 seconds after bootup. When triggered, the script attempts to find the reload reason. If the reload reason was due to a crash, the policy searches for the related crashinfo file and sends this information to a URL location specified by the user in the environment variable `_crash_reporter_url`. A CGI script, `interface_tm.cgi`, has been created to receive the URL from the `tm_crash_reporter.tcl` policy and save the crash information in a local database on the target URL machine.

A Perl CGI script, `interface_tm.cgi`, has been created and is designed to run on a machine that contains an HTTP server and is accessible by the device that runs the `tm_crash_reporter.tcl` policy. The `interface_tm.cgi` script parses the data passed into it from `tm_crash_reporter.tcl` and appends the crash information to a text file, creating a history of all crashes in the system. Additionally, detailed information on each crash is stored in three files in a crash database directory that is specified by the user. Another Perl CGI script, `crash_report_display.cgi`, has been created to display the information stored in the database created by the `interface_tm.cgi` script. The `crash_report_display.cgi` script should be placed on the same machine that contains `interface_tm.cgi`. The machine should be running a web browser such as Internet Explorer or Netscape. When the `crash_report_display.cgi` script is run, it displays the crash information in a readable format.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `tm_crash_reporter.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command to verify that the policy has been registered.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
  event manager policy tm_crash_reporter.tcl
end
show event manager policy registered
```

Running the `tm_fsys_usage.tcl` Sample Policy

This sample policy demonstrates the ability to periodically monitor disk space usage and report through syslog when configurable thresholds have been crossed.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `tm_fsys_usage.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered. If you had configured any of the optional environment variables that are used in the `tm_fsys_usage.tcl` policy, the **show event manager environment** command displays the configured variables.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy tm_fsys_usage.tcl
 end
show event manager policy registered
show event manager environment
```

Programming Policies with Tcl Sample Scripts Example

This section contains some of the sample policies that are included as EEM system policies. For more details about these policies, see the [EEM Event Detector Demo Examples, on page 1865](#).

Mandatory.go_ipsec.tcl Sample Policy

The following sample policy for the TestIPSecEncrypDecrypPkt Test.

```
::cisco::eem::event_register_gold card all testing_type monitoring test_name TestIPSecEncrypDecrypPkt consecutive_failure 6 platform_action 0 queue_priority last
#
# GOLD TestIPSecEncrypDecrypPkt Test TCL script
#
# March 2005, Hai Qiu
#
# Copyright (c) 2005-2007 by cisco Systems, Inc.
# All rights reserved.
#
#
# Register for TestIPSecEncrypDecrypPkt test even
# the elements for register the event
# card [all | card #]
# sub_card [all | sub_card #]
# severity_major | severity_minor | severity_normal default : severity_normal
# new_failure [true | false] default: dont_care
# testing_type [bootup | ondemand | schedule | monitoring]
# test_name [ test name ]
# test_id [ test # ]
# consecutive_failure [ consecutive_failure # ]
# platform_action [action_flag]
# action_flag [ 0 | 1 | 2 ]
# queue_priority [ normal | low | high | last] default: normal
#
# Note:
# 1: "card" element is required. If other elements are not specified,
# treat them as dont care, or default.
```

```

#
# 2: action_flag is platform specific. It is up to platform to
# determine what action need to be taken based on the value
# For Cat6k platform
# action_flag 0 : TCL script take action to reset card
# action_flag 1 : TCL script doesn't take action to reset card
# action_flag 2 : TCL script takes action to reset card for bootup diag
# when there is major error
# action_flag 3 : TCL script doesn't take action to reset card for
# bootup diag when there is major error
#
# 3: "queue_priority last" would guarantee this policy will be executed last
# if there are other EEM events in queue with queue priority other
# than "last"
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# 1. query the information of latest triggered eem event
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
puts "GOLD EEM TCL policy for TestIPSecEncrypDecrypPkt"
#set msg [format "array=%s", array names arr_einfo]
#puts "msg $msg"
#set msg $arr_einfo(msg)
set card $arr_einfo(card)
set sub_card $arr_einfo(sub_card)
#set overall_result $arr_einfo(overall_result)
#puts "GOLD event msg recieved: $card/$sub_card overall_result= $overall_result"
# 2. execute the user-defined config commands
if [catch {cli_open} result] {
    error $result $errorInfo
} else {
    array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
# Use "diagn action mod mod# test testname default" command
# for default platform action
if [catch {cli_exec $cli1(fd) "diagnostic action mod $card test TestIPSecEncrypD
ecrypPkt default"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}

```

ap_perf_test_base_cpu.tcl Sample Policy

The following sample policy measures the CPU performance of EEM policies.

```

::cisco::eem::event_register_appl sub_system 798 type 9999
#-----
# EEM policy used for measuring the cpu performance of EEM policies.
#
# July 2005, Cisco EEM team
#
# Copyright (c) 2005, 2006 by cisco Systems, Inc.

```

```

# All rights reserved.
#-----
###
### Input arguments:
###
### arg1 $iter          - current iteration count
###
### The following EEM environment variables are used:
###
### _perf_iterations (mandatory) - number of iterations over which we
###                               will run our measurement.
### Example:
### event manager environment _perf_iterations 100
###
### _perf_cmd1 (optional)      - optional non interactive cli command
###                               to be executed as part of the
###                               measurement test.
### Example:
### event manager environment _perf_cmd1 enable
###
### _perf_cmd2 (optional)      - optional non interactive cli command
###                               to be executed as part of the
###                               measurement test.
###                               To use _perf_cmd2, _perf_cmd1 MUST
###                               be defined.
### Example:
### event manager environment _perf_cmd2 show ver
###
### _perf_cmd3 (optional)      - optional non interactive cli command
###                               to be executed as part of the
###                               measurement test.
###                               To use _perf_cmd3, _perf_cmd1 MUST
###                               be defined.
### Example:
### event manager environment _perf_cmd3 show int counters protocol status
###
### Description:
### Iterate through _perf_iterations of this policy.
### It is up to the user to calculate the average
### execution time based on the system timestamps.
### Optional commands _perf_cmd1,
### _perf_cmd2 and _perf_cmd3 are executed if defined.
###
### A value of 100 is a good starting point.
###
### Outputs:
### Console output.
###
### Usage example:
### >conf t
### >service timestamps debug datetime msec
### >event manager environment _perf_iterations 100
### >event manager policy ap_perf_base_cpu.tcl
### >event manager policy no_perf_test_init.tcl
### >end
### 2d19h: %SYS-5-CONFIG_I: Configured from console by console
### >event manager run no_perf_test_init.tcl
###
### Oct 16 14:57:17.284: %SYS-5-CONFIG_I: Configured from console by console
### >event manager run no_perf_test_init.tcl
###
### Oct 16 19:32:02.772: %HA_EM-6-LOG:
### eem_policy/no_perf_test_init.tcl: EEM performance test start
### Oct 16 19:32:03.115: %HA_EM-6-LOG:

```

```

### eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 1
### Oct 16 19:32:03.467: %HA_EM-6-LOG:
### eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 2
### ...
### Oct 16 19:32:36.936: %HA_EM-6-LOG:
### eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 100
### Oct 16 19:32:36.936: %HA_EM-6-LOG:
### eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test end
###
### The user must calculate execution time and average time of execution.
### In this example, total time = 19:32:36.936 - 19:32:02.772 = 34.164
### Average script execution time = 341.64 milliseconds
###
# check if all the env variables we need exist
# If any of them doesn't exist, print out an error msg and quit
if ![info exists _perf_iterations] {
    set result \
        "Policy cannot be run: variable _perf_iterations has not been set"
    error $result $errorInfo
}
# ensure our target iteration count > 0
if {$_perf_iterations <= 0} {
    set result \
        "Policy cannot be run: variable _perf_iterations <= 0"
    error $result $errorInfo
}
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# query the event info
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
set iter $arr_einfo(data1)
set iter [expr $iter + 1]
# if _perf_cmd1 is defined
if {[info exists _perf_cmd1]} {
    # open the cli library
    if [catch {cli_open} result] {
        error $result $errorInfo
    } else {
        array set cli1 $result
    }
    # execute the comamnd defined in _perf_cmd1
    if [catch {cli_exec $cli1(fd) $_perf_cmd1} result] {
        error $result $errorInfo
    }
    # if _perf_cmd2 is defined
    if {[info exists _perf_cmd2]} {
        # execute the comamnd defined in _perf_cmd2
        if [catch {cli_exec $cli1(fd) $_perf_cmd2} result] {
            error $result $errorInfo
        } else {
            set cmd_output $result
        }
    }
    # if _perf_cmd3 is defined
    if {[info exists _perf_cmd3]} {
        # execute the comamnd defined in _perf_cmd3
        if [catch {cli_exec $cli1(fd) $_perf_cmd3} result] {
            error $result $errorInfo
        } else {

```

```

        set cmd_output $result
    }
}
# close the cli library
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}
}

# log a message
set msg [format "EEM performance test iteration %s" $iter]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
# use the context info from the previous run to determine when to end
if {$iter >= $_perf_iterations} {
    #log the final messages
    action_syslog priority info msg "EEM performance test end"
    if {$_cerrno != 0} {
        set result [format \
            "component=%s; subsys err=%s; posix err=%s;\n%s" \
            $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
        error $result
    }
    exit 0
}
# cause the next iteration to run
event_publish sub_system 798 type 9999 arg1 $iter
if {$_cerrno != 0} {
    set result [format \
        "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
}

```

tm_cli_cmd.tcl Sample Policy

The following sample policy runs a configurable CRON entry. The policy executes a configurable Cisco IOS CLI command and e-mails the results. An optional log file can be defined to which the output is appended with a timestamp.

```

::cisco::eem::event_register_timer cron name crontimer2 cron_entry $_cron_entry maxrun 240
#-----
# EEM policy that will periodically execute a cli command and email the
# results to a user.
#
# July 2005, Cisco EEM team
#
# Copyright (c) 2005 by cisco Systems, Inc.
# All rights reserved.
#-----
### The following EEM environment variables are used:
###
### _cron_entry (mandatory)           - A CRON specification that determines
###                                   when the policy will run. See the
###                                   IOS Embedded Event Manager
###                                   documentation for more information
###                                   on how to specify a cron entry.

```

```

### Example: _cron_entry          0-59/1 0-23/1 * * 0-7
###
### _log_file (mandatory without _email_....)
###                               - A filename to append the output to.
###                               If this variable is defined, the
###                               output is appended to the specified
###                               file with a timestamp added.
### Example: _log_file           bootflash:/my_file.log
###
### _email_server (mandatory without _log_file)
###                               - A Simple Mail Transfer Protocol (SMTP)
###                               mail server used to send e-mail.
### Example: _email_server       mailserver.example.com
###
### _email_from (mandatory without _log_file)
###                               - The address from which e-mail is sent.
### Example: _email_from         devtest@example.com
###
### _email_to (mandatory without _log_file)
###                               - The address to which e-mail is sent.
### Example: _email_to           engineering@example.com
###
### _email_cc (optional)
###                               - The address to which the e-mail must
###                               be copied.
### Example: _email_cc           manager@example.com
###
### _show_cmd (mandatory)
###                               - The CLI command to be executed when
###                               the policy is run.
### Example: _show_cmd           show version
###
# check if all required environment variables exist
# If any required environment variable does not exist, print out an error msg and quit
if {[info exists _log_file]} {
    if {[info exists _email_server]} {
        set result \
        "Policy cannot be run: variable _log_file or _email_server has not been set"
        error $result $errorMsg
    }
    if {[info exists _email_from]} {
        set result \
        "Policy cannot be run: variable _log_file or _email_from has not been set"
        error $result $errorMsg
    }
    if {[info exists _email_to]} {
        set result \
        "Policy cannot be run: variable _log_file ore _email_to has not been set"
        error $result $errorMsg
    }
    if {[info exists _email_cc]} {
        #_email_cc is an option, must set to empty string if not set.
        set _email_cc ""
    }
}
if {[info exists _show_cmd]} {
    set result \
    "Policy cannot be run: variable _show_cmd has not been set"
    error $result $errorMsg
}
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# query the event info and log a message
array set arr_einfo [event_reginfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \

```

```

        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
global timer_type timer_time_sec
set timer_type $arr_einfo(timer_type)
set timer_time_sec $arr_einfo(timer_time_sec)
# log a message
set msg [format "timer event: timer type %s, time expired %s" \
    $timer_type [clock format $timer_time_sec]]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
# 1. execute the command
if [catch {cli_open} result] {
    error $result $errorInfo
} else {
    array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
# save exact execution time for command
set time_now [clock seconds]
# execute command
if [catch {cli_exec $cli1(fd) $_show_cmd} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
    # format output: remove trailing router prompt
    regexp {\n*(.*\n)([^\n]*)$} $result dummy cmd_output
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}

# 2. log the success of the CLI command
set msg [format "Command \"%s\" executed successfully" $_show_cmd]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
# 3. if _log_file is defined, then attach it to the file
if {[info exists _log_file]} {
    # attach output to file
    if [catch {open $_log_file a+} result] {
        error $result
    }
    set fileD $result
    # save timestamp of command execution
    # (Format = 00:53:44 PDT Mon May 02 2005)
    set time_now [clock format $time_now -format "%T %Z %a %b %d %Y"]
    puts $fileD "%%% Timestamp = $time_now"
    puts $fileD $cmd_output
    close $fileD
}
# 4. if _email_server is defined send the email out
if {[info exists _email_server]} {
    set routename [info hostname]
    if {[string match "" $routename]} {

```

```

error "Host name is not configured"
}
if [catch {smtp_subst [file join $tcl_library email_template_cmd.tm]} \
result] {
error $result $errorInfo
}
if [catch {smtp_send_email $result} result] {
error $result $errorInfo
}
}
}

```

sl_intf_down.tcl Sample Policy

The following sample policy runs when a configurable syslog message is logged. The policy executes a configurable CLI command and e-mails the results.

```

::cisco::eem::event_register_syslog occurs 1 pattern $_syslog_pattern maxrun 90

#-----
# EEM policy to monitor for a specified syslog message.
# Designed to be used for syslog interface-down messages.
# When event is triggered, the given config commands will be run.
#
# July 2005, Cisco EEM team
#
# Copyright (c) 2005 by cisco Systems, Inc.
# All rights reserved.
#-----

### The following EEM environment variables are used:
###
### _syslog_pattern (mandatory)      - A regular expression pattern match string
###                                that is used to compare syslog messages
###                                to determine when policy runs
### Example: _syslog_pattern        .*UPDOWN.*FastEthernet0/0.*
###
### _email_server (mandatory)        - A Simple Mail Transfer Protocol (SMTP)
###                                mail server used to send e-mail.
### Example: _email_server          mailserver.example.com
###
### _email_from (mandatory)          - The address from which e-mail is sent.
### Example: _email_from            devtest@example.com
###
### _email_to (mandatory)            - The address to which e-mail is sent.
### Example: _email_to              engineering@example.com
###
### _email_cc (optional)             - The address to which the e-mail must
###                                be copied.
### Example: _email_cc              manager@example.com
###
### _config_cmd1 (optional)          - The first configuration command that
###                                is executed.
### Example: _config_cmd1           interface Ethernet1/0
###
### _config_cmd2 (optional)          - The second configuration command that
###                                is executed.
### Example: _config_cmd2           no shutdown
###

# check if all the env variables we need exist
# If any of them doesn't exist, print out an error msg and quit
if {[info exists _email_server]} {
    set result \

```



```

        "Policy cannot be run: variable _email_server has not been set"
        error $result $errorInfo
    }
    if {[info exists _email_from]} {
        set result \
            "Policy cannot be run: variable _email_from has not been set"
        error $result $errorInfo
    }
    if {[info exists _email_to]} {
        set result \
            "Policy cannot be run: variable _email_to has not been set"
        error $result $errorInfo
    }
    if {[info exists _email_cc]} {
        #_email_cc is an option, must set to empty string if not set.
        set _email_cc ""
    }
}

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

# 1. query the information of latest triggered eem event
array set arr_einfo [event_reqinfo]

if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

set msg $arr_einfo(msg)
set config_cmds ""

# 2. execute the user-defined config commands
if [catch {cli_open} result] {
    error $result $errorInfo
} else {
    array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
if [catch {cli_exec $cli1(fd) "config t"} result] {
    error $result $errorInfo
}

if {[info exists _config_cmd1]} {
    if [catch {cli_exec $cli1(fd) $_config_cmd1} result] {
        error $result $errorInfo
    }
    append config_cmds $_config_cmd1
}

if {[info exists _config_cmd2]} {
    if [catch {cli_exec $cli1(fd) $_config_cmd2} result] {
        error $result $errorInfo
    }
    append config_cmds "\n"
    append config_cmds $_config_cmd2
}

if [catch {cli_exec $cli1(fd) "end"} result] {
    error $result $errorInfo
}
}

```

```

if [catch {cli_close $cli(fd) $cli(tty_id)} result] {
    error $result $errorInfo
}

after 60000
# 3. send the notification email
set routername [info hostname]
if {[string match "" $routername]} {
    error "Host name is not configured"
}

if [catch {smtp_subst [file join $tcl_library email_template_cfg.tm]} result] {
    error $result $errorInfo
}

if [catch {smtp_send_email $result} result] {
    error $result $errorInfo
}

```

The following e-mail template file is used with the EEM sample policy above:

```

email_template_cfg.tm
Mailservername: $_email_server
From: $_email_from
To: $_email_to
Cc: $_email_cc
Subject: From router $routername: Periodic $_show_cmd Output
$cmd_output

```

Debugging Embedded Event Manager Policies Examples

The following examples show how to debug the CLI library and the SMTP library.

Debugging the CLI Library

The CLI library allows users to run CLI commands and obtain the output of commands in Tcl. An Embedded Event Manager **debug** command has been provided for users of this library. The command to enable CLI library debugging is **debug event manager tcl cli_library**. When enabled, this command displays all data that is passed in and read back from the TTY session that handles the CLI interactions. This data helps ensure users that the commands that they are passing to the CLI are valid.

Example of the debug event manager tcl cli_library Command

This example uses the sample policy `sl_intf_down.tcl`. When triggered, `sl_intf_down.tcl` passes a configuration command to the CLI through the CLI library. The command passed in below is **show event manager environment**. This command is not a valid command in configuration mode. Without the **debug** command enabled, the output is shown below:

```

00:00:57:sl_intf_down.tcl[0]:config_cmds are show eve man env
00:00:57:%SYS-5-CONFIG_I:Configured from console by vty0

```

Notice that with the output above the user would not know whether or not the command succeeded in the CLI. With the **debug event manager tcl cli_library** command enabled, the user sees the following:

```

01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : CTL : cli_open called.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson>
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson>enable
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson#

```

```

01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson#configure terminal
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : Enter configuration commands, one
per line. End with CNTL/Z.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson(config)#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson(config)#show event manager
environment
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT :
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : % Invalid input detected at '^'
marker.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson(config)#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson(config)#end
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : CTL : cli_close called.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson#exit
01:17:07: sl_intf_down.tcl[0]: config_cmds are show event manager environment
01:17:07: %SYS-5-CONFIG_I: Configured from console by vty0

```

The output above shows that **show event manager environment** is an invalid command in configuration mode. The IN keyword signifies all data passed in to the TTY through the CLI library. The OUT keyword signifies all data read back from the TTY through the CLI library. The CTL keyword signifies helper functions used in the CLI library. These helper functions are used to set up and remove connections to the CLI.

Debugging the SMTP Library

The SMTP library allows users to send e-mail messages to an SMTP e-mail server. An Embedded Event Manager **debug** command has been provided for users of this library. The command to enable SMTP library debugging is **debug event manager tcl smtp_library**. When enabled, this command displays all data that is passed in and read back from the SMTP library routines. This data helps ensure users that the commands that they are passing to the SMTP library are valid.

Example of the debug event manager tcl smtp_library Command

This example uses the sample policy `tm_cli_cmd.tcl`. When triggered, `tm_cli_cmd.tcl` runs the command **show event manager policy available system** through the CLI library. The result is then mailed to a user through the SMTP library. The output will help debug any issues related to using the SMTP library.

With the **debug event manager tcl smtp_library** command enabled, the users see the following on the console:

```

00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 220 XXXX.example.com ESMTP XXXX
1.1.0; Tue,
25 Jun 2002 14:20:39 -0700 (PDT)
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : HELO XXXX.example.com
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 XXXX.example.com Hello
XXXX.example.com [XXXX],
pleased to meet you
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : MAIL FROM:<XX@example.com>
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 <XX@example.com>... Sender
ok
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : RCPT TO:<XX@example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 <XX@example.com>... Recipient
ok
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : RCPT TO:<XX@example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 <XX@example.com>... Recipient
ok
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : DATA
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 354 Enter mail, end with "."
on a line by itself
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Date: 25 Jun 2002 14:35:00 UTC

```

```

00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Message-ID:
<20020625143500.2387058729877@XXXX.example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : From: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : To: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Cc: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Subject: From router nelson:
Periodic show eve man po ava system Output
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : No. Type Time Created
Name
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 1 system Fri May3 20:42:34
2002 pr_cdp_abort.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 2 system Fri May3 20:42:54
2002 pr_iprouting_abort.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 3 system Wed Apr3 02:16:33
2002 sl_intf_down.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 4 system Mon Jun24 23:34:16
2002 tm_cli_cmd.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 5 system Wed Mar27 05:53:15
2002 tm_crash_hist.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : nelson#
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write :
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : .
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 ADE90179 Message accepted
for delivery
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : QUIT
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 221 XXXX.example.com closing
connection

```

Tracing Tcl set Command Operations Example

Tcl is a flexible language. One of the flexible aspects of Tcl is that you can override commands. In this example, the Tcl `set` command is renamed as `_set` and a new version of the `set` command is created that displays a message containing the text “setting” and appends the scalar variable that is being set. This example can be used to trace all instances of scalar variables being set.

```

rename set _set
proc set {var args} {
    puts [list setting $var $args]
    uplevel _set $var $args
};

```

When this is placed in a policy, a message is displayed anytime a scalar variable is set, for example:

```
02:17:58: sl_intf_down.tcl[0]: setting test_var 1
```

RPC Event Detector Example

```

TCL script (rpccli.tcl):
::cisco::eem::event_register_rpc
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
proc run_cli {clist} {
    set rbuf ""
    if {[llength $clist] < 1} {
        return -code ok $rbuf
    }
    if {[catch {cli_open} result]} {
        return -code error $result
    }
}

```

```

    } else {
    array set cliarr $result
    }
    if {[catch {cli_exec $cliarr(fd) "enable"} result]} {
        return -code error $result
    }
    if {[catch {cli_exec $cliarr(fd) "term length 0"} result]} {
        return -code error $result
    }
    foreach cmd $clist {
    if {[catch {cli_exec $cliarr(fd) $cmd} result]} {
        return -code error $result
    }
    append rbuf $result
    }
    if {[catch {cli_close $cliarr(fd) $cliarr(tty_id)} result]} {
        puts "WARNING: $result"
    }
    return -code ok $rbuf
}
}
proc run_cli_interactive { clist } {
    set rbuf ""
    if {[length $clist] < 1} {
        return -code ok $rbuf
    }
    if {[catch {cli_open} result]} {
        return -code error $result
    }
    } else {
    array set cliarr $result
    }
    if {[catch {cli_exec $cliarr(fd) "enable"} result]} {
        return -code error $result
    }
    if {[catch {cli_exec $cliarr(fd) "term length 0"} result]} {
        return -code error $result
    }
    }
    foreach cmd $clist {
        array set sendexp $cmd
    if {[catch {cli_write $cliarr(fd) $sendexp(send)} result]} {
        return -code error $result
    }
    }
    foreach response $sendexp(responses) {
        array set resp $response
        if {[catch {cli_read_pattern $cliarr(fd) $resp(expect)} result]} {
            return -code error $result
        }
        if {[catch {cli_write $cliarr(fd) $resp(reply)} result]} {
            return -code error $result
        }
    }
    if {[catch {cli_read $cliarr(fd)} result]} {
        return -code error $result
    }
    append rbuf $result
    }
    if {[catch {cli_close $cliarr(fd) $cliarr(tty_id)} result]} {
        puts "WARNING: $result"
    }
    return -code ok $rbuf
}
}
array set arr_einfo [event_reqinfo]
set args $arr_einfo(argc)
set cmds [list]
for { set i 0 } { $i < $args } { incr i } {

```

```

set arg "arg${i}"
# Split each argument on the '^' character. The first element is
# the command, and each subsequent element is a prompt followed by
# a response to that prompt.
set cmdlist [split $arr_einfo($arg) "^"]
set cmdarr(send) [lindex $cmdlist 0]
set cmdarr(responses) [list]
if { [expr ([llength $cmdlist] - 1) % 2] != 0 } {
return -code 88
}
set cmdarr(responses) [list]
for { set j 1 } { $j < [llength $cmdlist] } { incr j 2 } {
set resp(expect) [lindex $cmdlist $j]
set resp(reply) [lindex $cmdlist [expr $j + 1]]
lappend cmdarr(responses) [array get resp]
}
lappend cmds [array get cmdarr]
}
set rc [catch {run_cli_interactive $cmds} output]
if { $rc != 0 } {
error $output $errorInfo
return -code 88
}
puts $output

```

Additional References

The following sections provide references related to writing Embedded Event Manager policies using Tcl.

Related Documents

Related Topic	Document Title
EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples	Cisco IOS Embedded Event Manager Command Reference
Embedded Event Manager overview	Embedded Event Manager Overview module.
Embedded Event Manager policy writing using the CLI	Writing Embedded Event Manager Policies Using the Cisco IOS CLI module
Embedded Resource Manager	Embedded Resource Manager module

MIBs

MIB	MIBs Link
CISCO-EMBEDDED-EVENT-MGR-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 200: Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI

Feature Name	Releases	Feature Information
Embedded Event Manager 4.0	15.2(5)E1	This feature was introduced and is supported only on c2960cx platform.



CHAPTER 90

Signed Tcl Scripts

The Signed Tcl Scripts feature allows you to create a certificate to generate a digital signature and sign a Tool Command Language (Tcl) script with that digital signature. This feature also allows you to work with existing scripts and certificates. The digital signature is verified for authentication and then run with trusted access to the Tcl interpreter. If the script does not contain the digital signature, the script may run in a limited mode for untrusted scripts, or may not run at all.

- [Prerequisites for Signed Tcl Scripts, on page 1889](#)
- [Restrictions for Signed Tcl Scripts, on page 1889](#)
- [Information About Signed Tcl Scripts, on page 1890](#)
- [How to Configure Signed Tcl Scripts, on page 1891](#)
- [Configuration Examples for Signed Tcl Script, on page 1904](#)
- [Additional References, on page 1908](#)
- [Feature Information for Signed Tcl Scripts, on page 1909](#)
- [Glossary, on page 1909](#)
- [Notices, on page 1910](#)

Prerequisites for Signed Tcl Scripts

For this feature to work, the Cisco public key infrastructure (PKI) configuration trustpoint commands must be enabled.

Restrictions for Signed Tcl Scripts

For this feature to work, you must be running the following:

- Cisco IOS Crypto image
- OpenSSL Version 0.9.7a or above
- Expect

Information About Signed Tcl Scripts

The Signed Tcl Scripts feature introduces security for the Tcl scripts. This feature allows you to create a certificate to generate a digital signature and sign a Tcl script with that digital signature. This certificate examines the Tcl scripts prior to running them. The script is checked for a digital signature from Cisco. In addition, third parties may also sign a script with a digital signature. You may wish to sign your own internally developed Tcl scripts or you could use a script developed by a third party. If the script contains the correct digital signature, it is believed to be authentic and runs with full access to the Tcl interpreter. If the script does not contain the digital signature, the script may be run in a limited mode, known as Safe Tcl mode, or may not run at all.

To create and use signed Tcl scripts, you should understand the following concepts:

Cisco PKI

Cisco PKI provides certificate management to support security protocols such as IP security (IPsec), secure shell (SSH), and secure socket layer (SSL). A PKI is composed of the following entities:

- Peers communicating on a secure network
- At least one certification authority (CA) that grants and maintains certificates
- Digital certificates, which contain information such as the certificate validity period, peer identity information, encryption keys that are used for secure communication, and the signature of the issuing CA
- An optional registration authority (RA) to offload the CA by processing enrollment requests
- A distribution mechanism (such as Lightweight Directory Access Protocol [LDAP] or HTTP) for certificate revocation lists (CRLs)

PKI provides you with a scalable, secure mechanism for distributing, managing, and revoking encryption and identity information in a secured data network. Every routing device participating in the secured communication is enrolled in the PKI in a process where the routing device generates a Rivest, Shamir, and Adelman (RSA) key pair (one private key and one public key) and has its identity validated by a trusted routing device (also known as a CA or trustpoint).

After each routing device enrolls in a PKI, every peer (also known as an end host) in a PKI is granted a digital certificate that has been issued by a CA. When peers must negotiate a secured communication session, they exchange digital certificates. Based on the information in the certificate, a peer can validate the identity of another peer and establish an encrypted session with the public keys contained in the certificate.

RSA Key Pair

An RSA key pair consists of a public key and a private key. When setting up your PKI, you must include the public key in the certificate enrollment request. After the certificate has been granted, the public key is included in the certificate so that peers can use it to encrypt data that is sent to the device. The private key is kept on the device and used both to decrypt the data sent by peers and to digitally sign transactions when negotiating with peers.

RSA key pairs contain a key modulus value. The modulus determines the size of the RSA key. The larger the modulus, the more secure the RSA key. However, keys with large modulus values take longer to generate, and encryption and decryption operations take longer with larger keys.

Certificate and Trustpoint

A certification authority (CA), also known as a trustpoint, manages certificate requests and issues certificates to participating network devices. These services (managing certificate requests and issuing certificates) provide centralized key management for the participating devices and are explicitly trusted by the receiver to validate identities and to create digital certificates. Before any PKI operations can begin, the CA generates its own public key pair and creates a self-signed CA certificate; thereafter, the CA can sign certificate requests and begin peer enrollment for the PKI.

You can use a CA provided by a third-party CA vendor, or you can use an internal CA, which is the Cisco Certificate Server.

How to Configure Signed Tcl Scripts

Generating a Key Pair

The key pair consists of a private key and a public key. The private key is intended to be kept private, accessible only to the creator. The public key is generated from the private key and is intended to be known to the public.

To generate a key pair, use the **openssl genrsa** command and then the **openssl rsa** command.

SUMMARY STEPS

1. **openssl genrsa -out** *private-key-file* *bit-length*
2. **ls -l**
3. **openssl rsa -in** *private-key-file* **-pubout -out** *public-key-file*
4. **ls -l**

DETAILED STEPS

Step 1 **openssl genrsa -out** *private-key-file* *bit-length*

This command generates a private key that is *bit-length* bits long and writes the key to the *private-key-file* file.

```
Host% openssl genrsa -out privkey.pem 2048
```

Example:

```
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
```

Step 2 **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

Example:

```
Host% ls -l
total 8
-rw-r--r--  1 janedoe eng12      1679 Jun 12 14:55 privkey.pem
```

The `privkey.pem` file contains the private key generated using the `openssl genrsa` command.

Step 3 `openssl rsa -in private-key-file -pubout -out public-key-file`

This command generates a public key based on the specified private key in the `private-key-file` file and writes the public key to the `public-key-file` file.

Example:

```
Host% openssl rsa -in privkey.pem -pubout -out pubkey.pem
writing RSA key
```

Step 4 `ls -l`

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

Example:

```
Host% ls -l
total 16
-rw-r--r--  1 janedoe eng12      1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe eng12       451 Jun 12 14:57 pubkey.pem
```

The `pubkey.pem` file contains the public key generated from the private key using the `openssl rsa` command.

Generating a Certificate

Perform this task to generate a certificate. To generate an X.509 certificate, use the `openssl req` command.

SUMMARY STEPS

1. `openssl req -new -x509 -key private-key-file -out certificate-file -days expiration-days`
2. `ls -l`

DETAILED STEPS

Step 1 `openssl req -new -x509 -key private-key-file -out certificate-file -days expiration-days`

This command creates an X.509 certificate, with full access to a private key that is stored in the `private-key-file` file, and stores the certificate in the `certificate-file` file. The certificate is configured to expire in `expiration-days` days.

To complete the command, enter the following Distinguished Name (DN) information when prompted:

- Country name
- State or province name
- Organization name
- Organizational unit name
- Common name
- Email address

At each prompt, text enclosed in square brackets indicates the default value that will be used if you do not enter a value before you press Enter.

This example shows how to create an X.509 certificate that has full access to the private key in the `privkey.pem` file. The certificate is written to the `cert.pem` file and will expire 1095 days after the creation date.

Example:

```
Host% openssl req -new -x509 -key privkey.pem -out cert.pem -days 1095
```

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value, If you enter '.', the field will be left blank.

Country Name (2 letter code) [GB]:**US**

State or Province Name (full name) [Berkshire]:**California**

Locality Name (eg, city) [Newbury]:**San Jose**

Organization Name (eg, company) [My Company Ltd]:**Cisco Systems, Inc.**

Organizational Unit Name (eg, section) []:**DEPT_ACCT**

Common Name (eg, your name or your server's hostname) []:**Jane**

Email Address []:**janedoe@company.com**

Step 2 **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

Example:

```
Host% ls -l
```

```
total 24
-rw-r--r--  1 janedoe eng12      1659 Jun 12 15:01 cert.pem
-rw-r--r--  1 janedoe eng12      1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe eng12       451 Jun 12 14:57 pubkey.pem
```

The `cert.pem` file contains the X.509 certificate created using the `openssl req` command.

Signing the Tcl Scripts

Perform this task to sign the Tcl scripts. You will need to sign the Tcl file and output in OpenSSL document in pkcs7 (PKCS#7) format.

To sign the Tcl file, use the **openssl smime** command with the **-sign** keyword.

SUMMARY STEPS

1. **openssl smime -sign -in *tcl-file* -out *signed-tcl-file* -signer *certificate-file* -inkey *private-key-file* -outform DER -binary**
2. **ls -l**

DETAILED STEPS

Step 1 **openssl smime -sign -in *tcl-file* -out *signed-tcl-file* -signer *certificate-file* -inkey *private-key-file* -outform DER -binary**

This command signs the Tcl filename *tcl-file* using the certificate stored in *certificate-file* and the private key stored in *private-key-file* file and then writes the signed Tcl file in DER PKCS#7 format to the *signed-tcl-file* file.

Example:

```
Host% openssl smime -sign -in hello -out hello.pk7 -signer cert.pem -inkey privkey.pem -outform DER -binary
```

Step 2 **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

Example:

```
Host% ls -l

total 40
-rw-r--r--  1 janedoe  eng12      1659 Jun 12 15:01 cert.pem
-rw-r--r--  1 janedoe  eng12         115 Jun 13 10:16 hello
-rw-r--r--  1 janedoe  eng12      1876 Jun 13 10:16 hello.pk7
-rw-r--r--  1 janedoe  eng12      1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe  eng12         451 Jun 12 14:57 pubkey.pem
```

The hello.pk7 file contains the signed Tcl file created by the **openssl smime** command from the unsigned Tcl file named hello and using the X.509 certificate in the cert.pem file.

Verifying the Signature

Perform this task to verify that the signature matches the data, use the **openssl smime** command with the **-verify** keyword. The original Tcl content must be provided in the input file, because the file does not have the original content.

SUMMARY STEPS

1. `openssl smime -verify -in signed-tcl-file -CAfile certificate-file -inform DER -content tcl-file`
2. `ls -l`

DETAILED STEPS

Step 1 `openssl smime -verify -in signed-tcl-file -CAfile certificate-file -inform DER -content tcl-file`

This command verifies the signed Tcl file stored in DER PKCS#7 format in *signed-tcl-file* using the trusted Certificate Authority (CA) certificates in *certificate-file* and then writes the detached content to the file *tcl-file*.

The following example shows how to verify the signature with the input file `hello.pk7`:

Example:

```
Host% openssl smime -verify -in hello.pk7 -CAfile cert.pem -inform DER -content hello

puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
Verification successful
```

Note The SSL command page describes `-in filename` as the input message to be encrypted or signed or the MIME message to be decrypted or verified. For more information, go to <http://www.openssl.org/>.

Step 2 `ls -l`

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

Example:

```
Host% ls -l

total 40
-rw-r--r--  1 janedoe eng12      1659 Jun 13 10:18 cert.pem
-rw-r--r--  1 janedoe eng12       115 Jun 13 10:17 hello
-rw-r--r--  1 janedoe eng12     1876 Jun 13 10:16 hello.pk7
-rw-r--r--  1 janedoe eng12     1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe eng12      451 Jun 12 14:57 pubkey.pem
```

The `hello` file contains the content detached from the signed Tcl file `hello.pk7` by running the `openssl smime` command with the `-verify` keyword. If the verification was successful, the signer's certificates are written to the X.509 certificate in the `cert.pem` file.

Converting the Signature into Nonbinary Data

Perform this task to convert the signature from binary to nonbinary data.

SUMMARY STEPS

1. `xxd -ps signed-tcl-file > nonbinary-signature-file`
2. Create a script that displays **#Cisco Tcl Signature V1.0** in the first line and inserts a comment character (#) at the beginning of each line of the input file and writes each line to a file whose name is formed by appending the text string “_sig” to the name of the input file.
3. Run the script, supplying the name of the file containing the nonbinary signature file (*nonbinary-signature-file*) as the input argument.
4. `ls -l`
5. `cat signed-tcl-file commented-nonbinary-signature-file > signed-tcl-script`
6. `cat signed-tcl-script`

DETAILED STEPS

Step 1 `xxd -ps signed-tcl-file > nonbinary-signature-file`

This command converts the signature in *signed-tcl-file* from binary to nonbinary data and stores it as a hexadecimal dump in the file *nonbinary-signature-file*.

Example:

```
Host% xxd -ps hello.pk7 > hello.hex
```

Step 2 Create a script that displays **#Cisco Tcl Signature V1.0** in the first line and inserts a comment character (#) at the beginning of each line of the input file and writes each line to a file whose name is formed by appending the text string “_sig” to the name of the input file.

In this example the `cat` command is used to display the contents of the script file named `my_append`.

Example:

```
Host% cat my_append

#!/usr/bin/env expect
set my_first {#Cisco Tcl Signature V1.0}
set newline {}
set my_file [lindex $argv 0]
set my_new_file ${my_file}_sig
set my_new_handle [open $my_new_file w]
set my_handle [open $my_file r]
puts $my_new_handle $newline
puts $my_new_handle $my_first
foreach line [split [read $my_handle] "\n"] {
    set new_line (#)
    append new_line $line
    puts $my_new_handle $new_line
}

close $my_new_handle
close $my_handle
```

Step 3 Run the script, supplying the name of the file containing the nonbinary signature file (*nonbinary-signature-file*) as the input argument.

In this example, the `my_append` script is run with the nonbinary signature file `hello.hex` specified as input. The output file will be named `hello.hex_sig`.

Example:

```
Host% my_append hello.hex
```

Step 4 **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

Example:

```
Host% ls -l
```

```
total 80
-rw-r--r--  1 janedoe eng12      1659 Jun 13 10:18 cert.pem
-rw-r--r--  1 janedoe eng12       115 Jun 13 10:17 hello
-rw-r--r--  1 janedoe eng12     3815 Jun 13 10:20 hello.hex
-rw-r--r--  1 janedoe eng12     3907 Jun 13 10:22 hello.hex_sig
-rw-r--r--  1 janedoe eng12     1876 Jun 13 10:16 hello.pk7
-rwxr--r--  1 janedoe eng12      444 Jun 13 10:22 my_append
-rw-r--r--  1 janedoe eng12     1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe eng12      451 Jun 12 14:57 pubkey.pem
```

The hello.hex file contains nonbinary data (stored as a hexadecimal dump) converted from the binary signature in the signed Tcl file hello.pk7. The my_append file contains the script that inserts a comment character at the beginning of each line of the input file. The hello.hex_sig file is the file created by running the my_append script on the nonbinary signature file.

Step 5 **cat signed-tcl-file commented-nonbinary-signature-file > signed-tcl-script**

This command appends the contents of the nonbinary signature file (*commented-nonbinary-signature-file*) to the signed Tcl file stored in DER PKCS#7 format (in the *signed-tcl-file* file). The concatenated output is written to the file *signed-tcl-script*.

Example:

```
Host% cat hello hello.hex_sig > hello.tcl
```

Step 6 **cat signed-tcl-script**

This command displays the contents of the file *signed-tcl-script*, which is the concatenation of content detached from the signed Tcl file and the nonbinary signature file.

Example:

```
Host% cat hello.tcl
```

```
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
#Cisco Tcl Signature V1.0
#3082075006092a864886f70d010702a08207413082073d020101310b3009
#06052b0e03021a0500300b06092a864886f70d010701a08204a13082049d
#30820385a003020102020100300d06092a864886f70d0101040500308195
#310b3009060355040613025553311330110603550408130a43616c69666f
#726e69613111300f0603550407130853616e204a6f7365311c301a060355
#040a1313436973636f2053797374656d732c20496e632e310e300c060355
#040b13054e53535447310d300b060355040313044a6f686e3121301f0609
#2a864886f70d01090116126a6c6175746d616e40636973636f2e636f6d30
```

```
#1e170d3037303631323232303134335a170d313030363131323230313433
#5a308195310b3009060355040613025553311330110603550408130a4361
#6c69666f726e69613111300f0603550407130853616e204a6f7365311c30
#1a060355040a1313436973636f2053797374656d732c20496e632e310e30
#0c060355040b13054e53535447310d300b060355040313044a6f686e3121
#301f06092a864886f70d01090116126a6c6175746d616e40636973636f2e
#636f6d30820122300d06092a864886f70d010105000382010f00308201
#0a0282010100a751eb5ec1f3009738c88a55987c07b759c36f3386342283
#67ea20a89d9483ae85e0c63eeded8ab3eb7a08006689f09136f172183665
#c971099ba54e77ab47706069bbefaaab8c50184396350e4cc870c4c3f477
#88c55c52e2cf411f05b59f0eac0678ff5cc238fdce2263a9fc6b6c244b8
#ffaead865c19c3d3172674a13b24c8f2c01dd8b1bd491c13e84e29171b85
#f28155d81ac8c69bb25ca23c2921d85fbf745c106e7aff93c72316cbc654
#4a34ea88174a8ba7777fa60662974e1fbac85a0f0aeac925dba6e5e850b8
#7caffce2fe8bb04b61b62f532b5893c081522d538005df81670b931b0ad0
#e1e76ae648f598a9442d5d0976e67c8d55889299147d0203010001a381f5
#3081f2301d0603551d0e04160414bc34132be952ff8b9e1af3b93140a255
#e54a667c3081c20603551d230481ba3081b78014bc34132be952ff8b9e1a
#f3b93140a255e54a667ca1819ba48198308195310b300906035504061302
#5553311330110603550408130a43616c69666f726e69613111300f060355
#0407130853616e204a6f7365311c301a060355040a1313436973636f2053
#797374656d732c20496e632e310e300c060355040b13054e53535447310d
#300b060355040313044a6f686e3121301f06092a864886f70d0109011612
#6a6c6175746d616e40636973636f2e636f6d820100300c0603551d130405
#30030101ff300d06092a864886f70d010104050003820101000c83c1b074
#6720929c9514af6d5df96f0a95639f047c40a607c83d8362507c58fa7f84
#aa699ec5e5bef61b2308297a0662c653ff446acfb6f5cb2dd162d939338
#a5e4d78a5c45021e5d4dbabb8784efbf50cab0f5125d164487b31f5cf933
#a9f68f82cd111cbab1739d7f372ec460a7946882874b0a0f22dd53acbd62
#a944a15e52e54a24341b3b8a820f23a5bc7ea7b2278bb56838b8a4051926
#af9c167274ff8449003a4e012bcf4f4b3e280f85209249a390d14df47435
#35efabc6720ea3d56803a84a2163db4478ae19d7d987ef6971c8312e280a
#aac0217d4fe620c6582a48faa8ea5e3726a99012e1d55f8d61b066381f77
#4158d144a43fb536c77d6a318202773082027302010130819b308195310b
#3009060355040613025553311330110603550408130a43616c69666f726e
#69613111300f0603550407130853616e204a6f7365311c301a060355040a
#1313436973636f2053797374656d732c20496e632e310e300c060355040b
#13054e53535447310d300b060355040313044a6f686e3121301f06092a86
#4886f70d01090116126a6c6175746d616e40636973636f2e636f6d020100
#300906052b0e03021a0500a081b1301806092a864886f70d010903310b06
#092a864886f70d010701301c06092a864886f70d010905310f170d303730
#3631333137313634385a302306092a864886f70d01090431160414372cb3
#72dc607990577fd0426104a42ee4158d2b305206092a864886f70d01090f
#31453043300a06082a864886f70d0307300e06082a864886f70d03020202
#0080300d06082a864886f70d0302020140300706052b0e030207300d0608
#2a864886f70d0302020128300d06092a864886f70d010101050004820100
#72db6898742f449b26d3ac18f43a1e7178834fb05ad13951bf042e127eea
#944b72b96f3b8ecf7eb52f3d0e383bf63651750223efe69eae04287c9dae
#b1f31209444108b31d34e46654c6c3cc10b5baba887825c224ec6f376d49
#00ff7ab2d9f88402dab9a2c2ab6aa3ecceef5a594bdc7d3a822c55e7daa
#aa0c2b067e06967f22a20e406fe21d9013ecc6bd9cd6d402c2749f8bea61
#9f8f87acfb9e10d6ce91502e34629adca6ee855419afafe6a8233333e14
#ad4c107901d1f2bca4d7ffaadddbc54192a25da662f8b8509782c76977b8
#94879453fbb00486ccc55f88db50fcc149bae066916b350089cde51a6483
#2ec14019611720fc5bbe2400f24225fc
```

Configuring the Device with a Certificate

Perform this task to configure the device with a certificate.

Before you begin

You must already have a Cisco IOS Crypto image; otherwise you cannot configure a certificate.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint *name***
4. **enrollment terminal**
5. **exit**
6. **crypto pki authenticate *name***
7. At the prompt, enter the base-encoded CA certificate.
8. **scripting tcl secure-mode**
9. **scripting tcl trustpoint *name***
10. **scripting tcl trustpoint untrusted {execute | safe-execute | terminate}**
11. **exit**
12. **tclsafe**

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 **configure terminal**

Enters global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 **crypto pki trustpoint *name***

Declares the device is to use the Certificate Authority (CA) *mytrust* and enters ca-trustpoint configuration mode.

Example:

```
Device(config)# crypto pki trustpoint mytrust
```

Step 4 **enrollment terminal**

Specifies manual cut-and-paste certificate enrollment. When this command is enabled, the device displays the certificate request on the console terminal, allowing you to enter the issued certificate on the terminal.

Example:

```
Device(ca-trustpoint)# enrollment terminal
```

Step 5 **exit**

Exits ca-trustpoint configuration mode and returns to global configuration mode.

Example:

```
Device(ca-trustpoint)# exit
```

Step 6 **crypto pki authenticate** *name*

Retrieves the CA certificate and authenticates it. Check the certificate fingerprint if prompted.

Note Because the CA signs its own certificate, you should manually authenticate the public key of the CA by contacting the CA administrator when you perform this command.

Example:

```
Device(config)# crypto pki authenticate mytrust
```

Step 7 At the prompt, enter the base-encoded CA certificate.**Example:**

```
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIIEuDCCA6CgAwIBAgIBADANBgkqhkiG9w0BAQQFADCBnjELMAkGA1UEBhMCVVMx
EzARBgNVBAGTCkNhbgG1mb3JuaWEwETAPBgNVBACTCFNhbiBkb3NlMRwwGgYDVQK
ExNDaXNjbyBTeXN0ZW1zLCBjbmuMQ4wDAYDVQQLEWVVOU1NURzEWMBQGA1UEAxMN
Sm9obiBMXYV0bWFubjEhMB8GCSqGSIb3DQEJARYSAmxhdXRtYW5AY21zY28uY29t
MB4XDTA2MTEwNzE3NTgwMVoXDTA5MTEwNjE3NTgwMVowgZ4xCzAJBgNVBAYTA1VT
MRMwEQYDVQQIEWpDYWxpZm9ybmlhMREwDwYDVQQHEWhTYW4gSm9zZTEcMBoGA1UE
ChMTQ21zY28uY21zZGVtcywgSW5jLjEOMAwGA1UECzMFTlNTVEcxFjAUBGNVBAMT
DUpvaG4gTGFlbG1hbm4xITAfBgkqhkiG9w0BCQEWEmpsYXV0bWFuQGNpc2NvLmNv
bTCCASiWdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALxtqTMCirMb+CdyWLuH
oWAM8CEJDwGgGL7MWBhoi3TSMd/ww2XBB9biBtd1H6jHsjCiOwAR5OorakwFpyf7
mvRj2PqJALs+Vn93VBKIG6rZU14+wdOx686BVddIZvEJQPbRoIYTz fazWV70aLMV
bd7/B7vF1SG1YK9y1tX9p9nZyZ0x47OAXetwOaGinvlG7VNuTXaASBLUjCRZsIlz
SBrXedBzG6+BuowmlFK45EYSlag5Rt9RGXXMBqzx91iyhrJ3zDDmkExa45yKJET
mAgDVMcpeteJtif47UDZJK30g4MbMyx/c8WghmJ54qRL9BZEPmDxMQkNP1018MA1
Q8sCAwEAaAOb/jCB+zAdBgNVHQ4EFgQU9/ToDvbMR3JfJ4xEa4X47oNFq5kwgcsG
A1UdIwSBwzCBwIAU9/ToDvbMR3JfJ4xEa4X47oNFq5mhgaSkgaEwgZ4xCzAJBgNV
BAYTA1VTMRMwEQYDVQQIEWpDYWxpZm9ybmlhMREwDwYDVQQHEWhTYW4gSm9zZTEc
MBoGA1UEChMTQ21zY28uY21zZGVtcywgSW5jLjEOMAwGA1UECzMFTlNTVEcxFjAU
BgNVBAMTDUpvaG4gTGFlbG1hbm4xITAfBgkqhkiG9w0BCQEWEmpsYXV0bWFuQGNp
c2NvLmNvbYIBADAMBgNVHRMEBTADAQH/MA0GCSqGSIb3DQEBAUAA4IBAQBtEs/4
MQeN9pT+XPCPg2ObQU8y2AadI+I34YK+fdHsFOh68hZhpstn2VpNEvkFXpADhgr
7DkNGtwTCLa481v70iNFViQVL+inNrZwWMxoTnUNCK7Hc5kHkXt6cj0mvsefVUzx
Xl70mauhESRvlmYWrJxSsrEILerZYsuv5HbFdand+/rErmp2HVyfdntLnKdSzmXJ
5lwE/Et2QtYNGor00BlLesowfslR3LhHi4wn+5is7mALgNw/NuTiUrlzH18OeB4m
wcpBIJsLaJu6ZUQ17IqdsWsa3fHd5qq0/k8P9z0YAYrf3+MFQr4ibvsYvHLO087
o2JslgW4qz34pqNh
Certificate has the following attributes:
    Fingerprint MD5: 1E327DBB 330936EB 2FB8EACB 4FD1133E
    Fingerprint SHA1: EE7FF9F4 05148842 B9D50FAC D76FDC9C E0703246
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
```

Step 8 **scripting tcl secure-mode**

Enables signature verification of the interactive Tcl scripts.

```
Device(config)# scripting tcl secure-mode
```

Step 9 **scripting tcl trustpoint name** *name*

Associates an existing configured trustpoint name with a certificate to verify Tcl scripts.

```
Device(config)# scripting tcl trustpoint name mytrust
```

Step 10 **scripting tcl trustpoint untrusted** {**execute** | **safe-execute** | **terminate**}

(Optional) Allows the interactive Tcl scripts to run regardless of the scripts failing in the signature check or in untrusted mode using one of the three keywords: **execute**, **safe-execute**, or **terminate**.

- **execute** --Executes Tcl scripts even if the signature verification fails. If the **execute** keyword is configured, signature verification is not at all performed.

Note Use of this keyword is usually not recommended because the signature verification is not at all performed.

The **execute** keyword is provided for internal testing purposes and to provide flexibility. For example, in a situation where a certificate has expired but the other configurations are valid and you want to work with the existing configuration, then you can use the execute keyword to work around the expired certificate.

- **safe-execute** --Allows the script to run in safe mode. You can use the tclsafe command and also enter the interactive Tcl shell safe mode to explore the safe mode Tcl commands that are available. In order to get a better understanding of what is available in this limited safe mode, use the tclsafe Exec command to explore the options.
- **terminate** --Stops any script from running and reverts to default behavior. The default policy is to terminate. When the last trustpoint name is removed, the untrusted action is also removed. The untrusted action cannot be entered until at least one trustpoint name is configured for Tcl.

The following example shows how to execute the Tcl script in safe mode using the **safe-execute** keyword when the signature verification fails.

```
Device(config)# scripting tcl trustpoint untrusted safe-execute
```

Step 11 **exit**

Exits global configuration mode and returns to privileged EXEC mode.

```
Device(config)# exit
```

Step 12 **tclsafe**

(Optional) Enables the interactive Tcl shell untrusted safe mode. This allows you to manually run Tcl commands from the Cisco command line interface in untrusted safe mode.

```
Device# tclsafe
```

Example:

Verifying the Trustpoint

To display the trustpoints that are configured in the device, use the **show crypto pki trustpoints** command.

SUMMARY STEPS

1. **enable**
2. **show crypto pki trustpoints**

DETAILED STEPS

Step 1 **enable**

This command enables privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 **show crypto pki trustpoints**

This command displays the trustpoints that are configured in the device.

Example:

```
Device# show
crypto pki trustpoints

Trustpoint mytrust:
  Subject Name:
  ea=janedoe@cisco.com
  cn=Jane
  ou=DEPT_ACCT
  o=Cisco
  l=San Jose
  st=California
  c=US
  Serial Number: 00
  Certificate configured.
```

Verifying the Signed Tcl Script

To verify that the Signed Tcl Script is properly running, use the **debug crypto pki transactions** command and the **tclsh** command.

SUMMARY STEPS

1. **enable**

2. `debug crypto pki transactions`
3. `tclsh flash:signed-tcl-file`

DETAILED STEPS

Step 1 `enable`

This command enables privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 `debug crypto pki transactions`

This command display debugging messages for the trace of interaction (message type) between the CA and the device.

Example:

```
Device# debug crypto pki transactions
Crypto PKI Trans debugging is on
```

Step 3 `tclsh flash:signed-tcl-file`

This command executes the Tcl script in Tcl shell.

Note The file should be a signed Tcl file.

Example:

```
Device# tclsh flash:hello.tcl

hello
argc = 0
argv =
argv0 = flash:hello.tcl
tcl_interactive = 0
device#
*Apr 21 04:46:18.563: CRYPTO_PKI: locked trustpoint mytrust, refcount is 1
*Apr 21 04:46:18.563: The PKCS #7 message has 0 verified signers.
*Apr 21 04:46:18.563: CRYPTO_PKI: Success on PKCS7 verify!
*Apr 21 04:46:18.563: CRYPTO_PKI: unlocked trustpoint mytrust, refcount is 0
```

What to Do Next

- To get an overview of Crypto, refer to the “Part 5: Implementing and Managing a PKI” section of the *Security Configuration Guide*.

Configuration Examples for Signed Tcl Script

Generating a Key Pair Example

The following example shows how to generate the key pair--a private key and a public key:

Generate a Private Key: Example

```
Host% openssl genrsa -out privkey.pem 2048
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
Host% ls -l
total 8
-rw-r--r--  1 janedoe eng12      1679 Jun 12 14:55 privkey.pem
Host%
```

Generate a Public Key from the Private Key

```
Host% openssl rsa -in privkey.pem -pubout -out pubkey.pem
writing RSA key
Host% ls -l
total 16
-rw-r--r--  1 janedoe eng12      1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe eng12       451 Jun 12 14:57 pubkey.pem
```

Generating a Certificate Example

The following example shows how to generate a certificate:

```
Host% openssl req -new -x509 -key privkey.pem -out cert.pem -days 1095
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value, If you enter '.', the field will be left
blank.
-----
Country Name (2 letter code) [GB]:US
State or Province Name (full name) [Berkshire]:California
Locality Name (eg, city) [Newbury]:San Jose
Organization Name (eg, company) [My Company Ltd]:Cisco Systems, Inc.
Organizational Unit Name (eg, section) []:DEPT_ACCT
Common Name (eg, your name or your server's hostname) []:Jane
Email Address []:janedoe@company.com
Host% ls -l
total 24
-rw-r--r--  1 janedoe eng12      1659 Jun 12 15:01 cert.pem
-rw-r--r--  1 janedoe eng12      1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe eng12       451 Jun 12 14:57 pubkey.pem
```


Signing the Tcl Scripts Example

The following example shows how to sign the Tcl scripts:

```
Host% openssl smime -sign -in hello -out hello.pk7 -signer cert.pem -inkey privkey.pem
-outform DER -binary
Host% ls -l
total 40
-rw-r--r--  1 janedoe  eng12      1659 Jun 12 15:01 cert.pem
-rw-r--r--  1 janedoe  eng12      115 Jun 13 10:16 hello
-rw-r--r--  1 janedoe  eng12      1876 Jun 13 10:16 hello.pk7
-rw-r--r--  1 janedoe  eng12      1679 Jun 12 14:55 privkey.pem
-rw-r--r--  1 janedoe  eng12      451 Jun 12 14:57 pubkey.pem
```

Verifying the Signature Example

The following example shows how to verify the signature:

```
Host% openssl smime -verify -in hello.pk7 -CAfile cert.pem -inform DER -content hello
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
Verification successful
```

Converting the Signature with Nonbinary Data Example

The following example shows how to convert the Tcl signature with nonbinary data:

```
#Cisco Tcl Signature V1.0
Then append the signature file to the end of the file.
Host% xxd -ps hello.pk7 > hello.hex
Host% cat my_append
#!/usr/bin/env expect
set my_first {#Cisco Tcl Signature V1.0}
set newline {}
set my_file [lindex $argv 0]
set my_new_file ${my_file}_sig
set my_new_handle [open $my_new_file w]
set my_handle [open $my_file r]

puts $my_new_handle $newline
puts $my_new_handle $my_first
foreach line [split [read $my_handle] "\n"] {
    set new_line {#}
    append new_line $line
    puts $my_new_handle $new_line
}

close $my_new_handle
close $my_handle
Host% my_append hello.hex
Host% ls -l
total 80
-rw-r--r--  1 janedoe  eng12      1659 Jun 12 15:01 cert.pem
-rw-r--r--  1 janedoe  eng12      115 Jun 13 10:16 hello
-rw-r--r--  1 janedoe  eng12      3815 Jun 13 10:20 hello.hex
```

Converting the Signature with Nonbinary Data Example

```

-rw-r--r-- 1 janedoe eng12      3907 Jun 13 10:22 hello.hex_sig
-rw-r--r-- 1 janedoe eng12      1876 Jun 13 10:16 hello.pk7
-rwxr--r-- 1 janedoe eng12      444 Jun 13 10:22 my_append
-rw-r--r-- 1 janedoe eng12      1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12      451 Jun 12 14:57 pubkey.pem
Host% cat hello hello.hex_sig > hello.tcl
Host% cat hello.tcl
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
#Cisco Tcl Signature V1.0
#3082075006092a864886f70d010702a08207413082073d020101310b3009
#06052b0e03021a0500300b06092a864886f70d010701a08204a13082049d
#30820385a003020102020100300d06092a864886f70d0101040500308195
#310b3009060355040613025553311330110603550408130a43616c69666f
#726e69613111300f0603550407130853616e204a6f7365311c301a060355
#040a1313436973636f2053797374656d732c20496e632e310e300c060355
#040b13054e53535447310d300b060355040313044a6f686e3121301f0609
#2a864886f70d01090116126a6c6175746d616e40636973636f2e636f6d30
#1e170d3037303631323232303134335a170d313030363131323230313433
#5a308195310b3009060355040613025553311330110603550408130a4361
#6c69666f726e69613111300f0603550407130853616e204a6f7365311c30
#1a060355040a1313436973636f2053797374656d732c20496e632e310e30
#0c060355040b13054e53535447310d300b060355040313044a6f686e3121
#301f06092a864886f70d01090116126a6c6175746d616e40636973636f2e
#636f6d30820122300d06092a864886f70d010105000382010f00308201
#0a0282010100a751eb5ec1f3009738c88a55987c07b759c36f3386342283
#67ea20a89d9483ae85e0c63eeded8ab3eb7a08006689f09136f172183665
#c971099ba54e77ab47706069bbefaaab8c50184396350e4cc870c4c3f477
#88c55c52e2c411f05b59f0eaec0678ff5cc238fdce2263a9fc6b6c244b8
#fffaead865c19c3d3172674a13b24c8f2c01dd8b1bd491c13e84e29171b85
#f28155d81ac8c69bb25ca23c2921d85fbf745c106e7aff93c72316cbc654
#4a34ea88174a8ba7777fa60662974e1fbac85a0f0aeac925dba6e5e850b8
#7caffce2fe8bb04b61b62f532b5893c081522d538005df81670b931b0ad0
#e1e76ae648f598a9442d5d0976e67c8d55889299147d0203010001a381f5
#3081f2301d0603551d0e04160414bc34132be952ff8b9e1af3b93140a255
#e54a667c3081c20603551d230481ba3081b78014bc34132be952ff8b9e1a
#f3b93140a255e54a667ca1819ba48198308195310b300906035504061302
#5553311330110603550408130a43616c69666f726e69613111300f060355
#0407130853616e204a6f7365311c301a060355040a1313436973636f2053
#797374656d732c20496e632e310e300c060355040b13054e53535447310d
#300b060355040313044a6f686e3121301f06092a864886f70d0109011612
#6a6c6175746d616e40636973636f2e636f6d820100300c0603551d130405
#30030101ff300d06092a864886f70d010104050003820101000c83c1b074
#6720929c9514af6d5df96f0a95639f047c40a607c83d8362507c58fa7f84
#aa699ec5e5bef61b2308297a0662c653ff446acfb6f5cb2dd162d939338
#a5e4d78a5c45021e5d4dbabb8784efbf50cab0f5125d164487b31f5cf933
#a9f68f82cd111cbab1739d7f372ec460a7946882874b0a0f22dd53acbd62
#a944a15e52e54a24341b3b8a820f23a5bc7ea7b2278bb56838b8a4051926
#af9c167274ff8449003a4e012bcf4f4b3e280f85209249a390d14df47435
#35efabce720ea3d56803a84a2163db4478ae19d7d987ef6971c8312e280a
#aac0217d4fe620c6582a48faa8ea5e3726a99012e1d55f8d61b066381f77
#4158d144a43fb536c77d6a318202773082027302010130819b308195310b
#3009060355040613025553311330110603550408130a43616c69666f726e
#69613111300f0603550407130853616e204a6f7365311c301a060355040a
#1313436973636f2053797374656d732c20496e632e310e300c060355040b
#13054e53535447310d300b060355040313044a6f686e3121301f06092a86
#4886f70d01090116126a6c6175746d616e40636973636f2e636f6d020100
#300906052b0e03021a0500a081b1301806092a864886f70d010903310b06
#092a864886f70d010701301c06092a864886f70d010905310f170d303730
#3631333137313634385a302306092a864886f70d01090431160414372cb3
#72dc607990577fd0426104a42ee4158d2b305206092a864886f70d01090f

```

```
#31453043300a06082a864886f70d0307300e06082a864886f70d03020202
#0080300d06082a864886f70d0302020140300706052b0e030207300d0608
#2a864886f70d0302020128300d06092a864886f70d010101050004820100
#72db6898742f449b26d3ac18f43a1e7178834fb05ad13951bf042e127eea
#944b72b96f3b8ecf7eb52f3d0e383bf63651750223efe69eae04287c9dae
#b1f31209444108b31d34e46654c6c3cc10b5baba887825c224ec6f376d49
#00ff7ab2d9f88402dab9a2c2ab6aa3ecceef5a594bdc7d3a822c55e7daa
#aa0c2b067e06967f22a20e406fe21d9013ecc6bd9cd6d402c2749f8bea61
#9f8f87acfb9e10d6ce91502e34629adca6ee855419afafe6a823333e14
#ad4c107901d1f2bca4d7ffaaddbc54192a25da662f8b8509782c76977b8
#94879453fbb00486ccc55f88db50fcc149bae066916b350089cde51a6483
#2ec14019611720fc5bbe2400f24225fc
```

Configuring the Device with a Certificate Example

The following example shows how to configure the device with a certificate:

```
crypto pki trustpoint mytrust
  enrollment terminal
!
!
crypto pki authentication mytrust
crypto pki certificate chain mytrust
certificate ca 00
  308204B8 308203A0 A0030201 02020100 300D0609 2A864886 F70D0101 04050030
  819E310B 30090603 55040613 02555331 13301106 03550408 130A4361 6C69666F
  726E6961 3111300F 06035504 07130853 616E204A 6F736531 1C301A06 0355040A
  13134369 73636F20 53797374 656D732C 20496E63 2E310E30 0C060355 040B1305
  4E535354 47311630 14060355 0403130D 4A6F686E 204C6175 746D616E 6E312130
  1F06092A 864886F7 0D010901 16126A6C 6175746D 616E4063 6973636F 2E636F6D
  301E170D 30363131 31373137 35383031 5A170D30 39313131 36313735 3830315A
  30819E31 0B300906 03550406 13025553 31133011 06035504 08130A43 616C6966
  6F726E69 61311130 0F060355 04071308 53616E20 4A6F7365 311C301A 06035504
  0A131343 6973636F 20537973 74656D73 2C20496E 632E310E 300C0603 55040B13
  054E5353 54473116 30140603 55040313 0D4A6F68 6E204C61 75746D61 6E6E3121
  301F0609 2A864886 F70D0109 0116126A 6C617574 6D616E40 63697363 6F2E636F
  6D308201 22300D06 092A8648 86F70D01 01010500 0382010F 00308201 0A028201
  0100BC6D A933028A B31BF827 7258BB87 A1600CF0 21090F04 2080BECC 5818688B
  74D231DF F0C365C1 07D6E206 D7651FA8 C7B230A2 3B0011E4 EA2B6A4C 1F3F27FB
  9AF449D8 FA8900BB 3E567F77 5412881B AAD9525E 3EC1D3B1 EBCE8155 D74866F1
  0940F6D1 3A2613CD F6B3595E F468B315 6DDEF077 BBC5D521 B560AF72 D6D5FDA7
  D9D9C99D 31E3B380 5DEB7039 A1A29EF9 46ED536E 4D768048 12D48C24 59B08973
  481AD75D E741CD9E BE06EA16 9B514AE3 91184A56 A0E51B7D 4465D730 1AB3C7DD
  62CA1AC9 DF30C39A 41316B8E 72289113 98080354 C7297AD7 89B627F8 ED40D924
  ADF48383 1B332C7F 73C58686 6279E2A4 4BF41644 3E60F131 090D3F5D 25F0C025
  43CB0203 010001A3 81FE3081 FB301D06 03551D0E 04160414 F7F4E80E F6CC4772
  5F278C44 6B85F8EE 8345AB99 3081CB06 03551D23 0481C330 81C08014 F7F4E80E
  F6CC4772 5F278C44 6B85F8EE 8345AB99 A181A4A4 81A13081 9E310B30 09060355
  04061302 55533113 30110603 55040813 0A43616C 69666F72 6E696131 11300F06
  03550407 13085361 6E204A6F 7365311C 301A0603 55040A13 13436973 636F2053
  79737465 6D732C20 496E632E 310E300C 06035504 0B13054E 53535447 31163014
  06035504 03130D4A 6F686E20 4C617574 6D616E6E 3121301F 06092A86 4886F70D
  01090116 126A6C61 75746D61 6E406369 73636F2E 636F6D82 0100300C 0603551D
  13040530 030101FF 300D0609 2A864886 F70D0101 04050003 82010100 6D12CFF8
  31078DF6 94FE5CF0 8F83639B 414F32D8 069D23E2 37E182BE 7C31EC14 E87AF216
  61A6CCD3 37656934 4BE4157A 400E182B EC390D1A DC130A56 B8F35BFB D2234556
  24152FE8 A736B670 58CC684E 750D08AE C7739907 917B7A72 3D26BEC7 9F554CF1
  5E5EF499 ABA11124 55966616 AC9C52B2 B1082DEA D962CBAF E476C575 A9DDFBFA
  C4AE63F6 1D5C9F76 7B4B9CA7 52CE65C9 E65C04FC 4B7642D6 0D1A8AF4 38194B7A
  CA307EC9 51DCB847 8B8C27FB 98ACEE60 0B80DC3F 36E4E252 BD731F5F 0E781E26
  C1CA4120 9B0B689B BA654250 97B22A76 CC126B77 C7779AAA D3F93C3F DCF46006
  2B7F7F8C 150AF889 BBEC62F1 E53B4F3B A3626CD6 05B8AB3D F8A6A361
```

```

quit
archive
 log config
scripting tcl trustpoint name mytrust
scripting tcl secure-mode
!
!
end

```

Additional References

The following sections provide references related to writing EEM policies Using the Cisco IOS CLI.

Related Documents

Related Topic	Document Title
EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples	Cisco IOS Embedded Event Manager Command Reference
Embedded Event Manager overview	Embedded Event Manager Overview module
Embedded Event Manager policy writing using Tcl	Writing Embedded Event Manager Policies Using Tcl module
Configuring enhanced object tracking	Configuring Enhanced Object Tracking module

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	--

MIBs

MIB	MIBs Link
CISCO-EMBEDDED-EVENT-MGR-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	--

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature Information for Signed Tcl Scripts

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 201: Feature Information for Signed Tcl Scripts

Feature Name	Releases	Feature Information
Signed Tcl Scripts	15.2(5)E1	This feature was introduced and is supported only on c2960cx.

Glossary

CA--certification authority. Service responsible for managing certificate requests and issuing certificates to participating IPsec network devices. This service provides centralized key management for the participating devices and is explicitly trusted by the receiver to validate identities and to create digital certificates.

certificates--Electronic documents that bind a user's or device's name to its public key. Certificates are commonly used to validate a digital signature.

CRL--certificate revocation list. Electronic document that contains a list of revoked certificates. The CRL is created and digitally signed by the CA that originally issued the certificates. The CRL contains dates for when the certificate was issued and when it expires. A new CRL is issued when the current CRL expires.

IPsec--IP security

peer certificate--Certificate presented by a peer, which contains the peer's public key and is signed by the trustpoint CA.

PKI--public key infrastructure. System that manages encryption keys and identity information for components of a network that participate in secured communications.

RA--registration authority. Server that acts as a proxy for the CA so that CA functions can continue when the CA is offline. Although the RA is often part of the CA server, the RA could also be an additional application, requiring an additional device to run it.

RSA keys--Public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adleman. An RSA key pair (a public and a private key) is required before you can obtain a certificate for your device.

SHA1--Secure Hash Algorithm 1

SSH--secure shell

SSL--secure socket layer

Notices

The following notices pertain to this software license.

OpenSSL Open SSL Project

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product includes software written by Tim Hudson (tjh@cryptsoft.com).

License Issues

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License:

Copyright © 1998-2007 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)".

4. The names “OpenSSL Toolkit” and “OpenSSL Project” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called “OpenSSL” nor may “OpenSSL” appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:

“This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)”.

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License:

Copyright © 1995-1998 Eric Young (eay@cryptsoft.com). All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com).

The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young’s, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

“This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)”.

The word ‘cryptographic’ can be left out if the routines from the library being used are not cryptography-related.

1. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: “This product includes software written by Tim Hudson (tjh@cryptsoft.com)”.

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution license [including the GNU Public License].



CHAPTER 91

EEM CLI Library Command Extensions

All command-line interface (CLI) library command extensions belong to the `::cisco::eem` namespace.

This library provides users the ability to run CLI commands and get the output of the commands in Tcl. Users can use commands in this library to spawn an exec and open a virtual terminal channel to it, write the command to execute to the channel so that the command will be executed by exec, and read back the output of the command.

There are two types of CLI commands: interactive commands and non-interactive commands.

For interactive commands, after the command is entered, there will be a "Q&A" phase in which the device will ask for different user options, and the user is supposed to enter the answer for each question. Only after all the questions have been answered properly will the command run according to the user's options until completion.

For noninteractive commands, once the command is entered, the command will run to completion. To run different types of commands using an EEM script, different CLI library command sequences should be used, which are documented in the "Using the CLI Library to Run a Noninteractive Command" section and in the "Using the CLI Library to Run an Interactive Command" section in the `cli_write` Tcl command.

The vty lines are allocated from the pool of vty lines that are configured using the `line vty` CLI configuration command. EEM will use a vty line when a vty line is not being used by EEM and there are available vty lines. EEM will also use a vty line when EEM is already using a vty line and there are three or more vty lines available. Be aware that the connection will fail when fewer than three vty lines are available, preserving the remaining vty lines for Telnet use.

Your release may support XML-PI. For details about the XML-PI support, the new CLI library command extensions, and some examples of how to implement XML-PI, see EEM CLI Library XML-PI Support.

- [cli_close](#), on page 1914
- [cli_exec](#), on page 1914
- [cli_get_ttyname](#), on page 1915
- [cli_open](#), on page 1915
- [cli_read](#), on page 1916
- [cli_read_drain](#), on page 1916
- [cli_read_line](#), on page 1917
- [cli_read_pattern](#), on page 1917
- [cli_run](#), on page 1918
- [cli_run_interactive](#), on page 1919
- [cli_write](#), on page 1920

- [EEM 4.0 CLI Library XML-PI Support, on page 1923](#)
- [EEM CLI Library XML-PI Support, on page 1923](#)

cli_close

Closes the exec process and releases the vty and the specified channel handler connected to the command-line interface (CLI).

Syntax

```
cli_close fd tty_id
```

Arguments

fd	(Mandatory) The CLI channel handler.
tty_id	(Mandatory) The TTY ID returned from the cli_open command extension.

Result String

None

Set_cerrno

Cannot close the channel.

cli_exec

Writes the command to the specified channel handler to execute the command. Then reads the output of the command from the channel and returns the output.

Syntax

```
cli_exec fd cmd
```

Arguments

fd	(Mandatory) The command-line interface (CLI) channel handler.
cmd	(Mandatory) The CLI command to execute.

Result String

The output of the CLI command executed.

Set_cerrno

Error reading the channel.

cli_get_ttyname

Returns the real and pseudo TTY names for a given TTY ID.

Syntax

```
cli_get_ttyname tty_id
```

Arguments

tty_id	(Mandatory) The TTY ID returned from the cli_open command extension.
--------	-----------------------------------------------------------------------------

Result String

```
pty %s tty %s
```

Set_cerrno

None

cli_open

Allocates a vty, creates an EXEC command-line interface (CLI) session, and connects the vty to a channel handler. Returns an array including the channel handler.



Note Each call to **cli_open** initiates a Cisco IOS EXEC session that allocates a Cisco IOS vty line. The vty remains in use until the **cli_close** routine is called. The vty lines are allocated from the pool of vty lines that are configured using the **line vty** CLI configuration command. EEM will use a vty line when a vty line is not being used by EEM and there are available vty lines. EEM will also use a vty line when EEM is already using a vty line and there are three or more vty lines available. Be aware that the connection will fail when fewer than three vty lines are available, preserving the remaining vty lines for Telnet use

Syntax

```
cli_open
```

Arguments

None

Result String

```
"tty_id {%s} pty {%d} tty {%d} fd {%d}"
```

Event Type	Description
tty_id	TTY ID.
pty	PTY device name.
tty	TTY device name.
fd	CLI channel handler.

Set_cerrno

- Cannot get pty for EXEC.
- Cannot create an EXEC CLI session.
- Error reading the first prompt.

cli_read

Reads the command output from the specified command-line interface (CLI) channel handler until the pattern of the device prompt occurs in the contents read. Returns all the contents read up to the match.

Syntax

```
cli_read fd
```

Arguments

fd	(Mandatory) The CLI channel handler.
-----------	--------------------------------------

Result String

All the contents read.

Set_cerrno

Cannot get device name.



Note This Tcl command extension will block waiting for the device prompt to show up in the contents read.

cli_read_drain

Reads and drains the command output of the specified command-line interface (CLI) channel handler. Returns all the contents read.

Syntax

```
cli_read_drain fd
```

Arguments

fd	(Mandatory) The CLI channel handler.
----	--------------------------------------

Result String

All the contents read.

Set_cerrno

None

cli_read_line

Reads one line of the command output from the specified command-line interface (CLI) channel handler. Returns the line read.

Syntax

```
cli_read_line fd
```

Arguments

fd	(Mandatory) The CLI channel handler.
----	--------------------------------------

Result String

The line read.

Set_cerrno

None



Note This Tcl command extension will block waiting for the end of line to show up in the contents read.

cli_read_pattern

Reads the command output from the specified command-line interface (CLI) channel handler until the pattern that is to be matched occurs in the contents read. Returns all the contents read up to the match.



Note The pattern matching logic attempts a match by looking at the command output data as it is delivered from the Cisco IOS command. The match is always done on the most recent 256 characters in the output buffer unless there are fewer characters available, in which case the match is done on fewer characters. If more than 256 characters in the output buffer are required for the match to succeed, the pattern will not match.

Syntax

```
cli_read_pattern fd ptn
```

Arguments

fd	(Mandatory) The CLI channel handler.
ptn	(Mandatory) The pattern to be matched when reading the command output from the channel.

Result String

All the contents read.

Set_cerrno

None



Note This Tcl command extension will block waiting for the specified pattern to show up in the contents read.

cli_run

Iterates over the items in the clist and assumes that each one is a command-line-interface (CLI) command to be executed in the enable mode. On success, returns the output of all executed commands and on failure, returns error from the failure.

Syntax

```
cli_run clist
```

Arguments

clist	(Mandatory) The list of commands to be executed.
-------	--------------------------------------------------

Result String

Output of all the commands that are executed or an error message.

Set _cerrno

None.

Sample Usage

The following example shows how to use the **cli_run** command extension.

```
set clist [list {sh run} {sh ver} {sh event man pol reg}]
cli_run { clist }
```

cli_run_interactive

Provides a sublist to the clist which has three items. On success, returns the output of all executed commands and on failure, returns error from the failure. Also uses arrays when possible as a way of making things easier to read later by keeping expect and reply separated.

Syntax

```
cli_run_interactive clist
```

Arguments

clist	<p>(Mandatory) List of three items:</p> <ul style="list-style-type: none"> • command– Command to be executed • expect– A regular expression pattern match for the expected reply prompt • responses– A list of possible responses to the reply prompt constructed as an array of two items: <ul style="list-style-type: none"> • expect– A regular expression pattern match for a possible reply prompt • reply- A reply for that expected prompt
-------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Result String

Output of all the commands that are executed or an error message. As each command is executed its output is appended to a result variable. Upon exhaustion of the input list, the CLI channel is closed and the aggregate result is returned.

Set _cerrno

None.

Sample Usage

The following example shows how to clear counters for interface fa0/0 use the cli_run_interactive command extension.

```

set cmdarr(command) "clear counters fa0/0"
set cmdarr(responses) [list]
set resps(expect) {[confirm]}
set resps(reply) "y"
lappend cmdarr(responses) [array get resps]
set rc [catch {cli_run_interactive [list [array get cmdarr]]} result]

```

Possible errors raised include:

- cannot get pty for exec
- cannot spawn exec
- error reading the first prompt
- error reading the channel
- cannot close channel

cli_write

Writes the command that is to be executed to the specified CLI channel handler. The CLI channel handler executes the command.

Syntax

```
cli_write fd cmd
```

Arguments

fd	(Mandatory) The CLI channel handler.
cmd	(Mandatory) The CLI command to execute.

Result String

None

Set_cerrno

None

Sample Usage

As an example, use configuration CLI commands to bring up Ethernet interface 1/0:

```

if [catch {cli_open} result] {
puts stderr $result
exit 1
} else {
array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
puts stderr $result
exit 1
}

```



```

}
if [catch {cli_exec $cli1(fd) "config t"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "interface Ethernet1/0"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "no shut"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "end"} result] {
puts stderr $result
exit 1
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} } result] {
puts stderr $result
exit 1
}

```

Using the CLI Library to Run a Noninteractive Command

To run a noninteractive command, use the **cli_exec** command extension to issue the command, and then wait for the complete output and the device prompt. For example, the following shows the use of configuration CLI commands to bring up Ethernet interface 1/0:

```

if [catch {cli_open} result] {
error $result $errorInfo
} else {
set fd $result
}
if [catch {cli_exec $fd "en"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "config t"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "interface Ethernet1/0"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "no shut"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "end"} result] {
error $result $errorInfo
}
if [catch {cli_close $fd} result] {
error $result $errorInfo
}
}

```

Using the CLI Library to Run an Interactive Command

To run interactive commands, three phases are needed:

- Phase 1: Issue the command using the **cli_write** command extension.
- Phase 2: Q&A Phase. Use the **cli_read_pattern** command extension to read the question (the regular pattern that is specified to match the question text) and the **cli_write** command extension to write back the answers alternately.

- Phase 3: Noninteractive phase. All questions have been answered, and the command will run to completion. Use the **cli_read** command extension to wait for the complete output of the command and the device prompt.

For example, use CLI commands to do squeeze bootflash: and save the output of this command in the Tcl variable `cmd_output`.

```

if [catch {cli_open} result] {
error $result $errorInfo
} else {
array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
error $result $errorInfo
}

# Phase 1: issue the command
if [catch {cli_write $cli1(fd) "squeeze bootflash:"} result] {
error $result $errorInfo
}

# Phase 2: Q&A phase
# wait for prompted question:
# All deleted files will be removed. Continue? [confirm]
if [catch {cli_read_pattern $cli1(fd) "All deleted"} result] {
error $result $errorInfo
}
# write a newline character
if [catch {cli_write $cli1(fd) "\n"} result] {
error $result $errorInfo
}
# wait for prompted question:
# Squeeze operation may take a while. Continue? [confirm]
if [catch {cli_read_pattern $cli1(fd) "Squeeze operation"} result] {
error $result $errorInfo
}
# write a newline character
if [catch {cli_write $cli1(fd) "\n"} result] {
error $result $errorInfo
}

# Phase 3: noninteractive phase
# wait for command to complete and the router prompt
if [catch {cli_read $cli1(fd) } result] {
error $result $errorInfo
} else {
set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
error $result $errorInfo
}

```

The following example causes a device to be reloaded using the CLI **reload** command. Note that the EEM **action_reload** command accomplishes the same result in a more efficient manner, but this example is presented to illustrate the flexibility of the CLI library for interactive command execution.

```

# 1. execute the reload command
if [catch {cli_open} result] {
error $result $errorInfo
} else {
array set cli1 $result

```

```

}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
if [catch {cli_write $cli1(fd) "reload"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_read_pattern $cli1(fd) ".*(System configuration has been modified. Save\\? \\[yes/no\\]: )"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_write $cli1(fd) "no"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_read_pattern $cli1(fd) ".*(Proceed with reload\\? \\[confirm\\])"} result]
{
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_write $cli1(fd) "y"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}
}

```

EEM 4.0 CLI Library XML-PI Support

EEM CLI Library XML-PI Support

XML Programmatic Interface (XML-PI) was introduced in Cisco IOS Release 12.4(22)T. XML-PI provides a programmable interface which encapsulates IOS command-line interface (CLI) show commands in XML format in a consistent way across different Cisco products. Customers using XML-PI will be able to parse IOS show command output from within Tcl scripts using well-known keywords instead of having to depend on the use of regular expression support to "screen-scrape" output.

The benefit of using the XML-PI command extensions is to facilitate the extraction of specific output information that is generated using a CLI **show** command. Most show commands return many fields within the output and currently a regular expression has to be used to extract specific information that may appear in the middle of a line. XML-PI support provides a set of Tcl library functions to facilitate the parsing of output from the IOS CLI format extension in the form of:

```

show
<
show-command
> | format
{

```

```
spec-file  
}
```

where a spec-file is a concatenation of all Spec File Entries (SFE) for each **show** command currently supported. As part of the XML-PI project a default spec-file will be included in the IOS Release 12.4(22)T images. The default spec-file will have a small set of commands and the SFE for the commands will have a subset of the possible tags. If no spec-file is provided with the format command, the default spec-file is used.

For more general details about XML-PI, see the "XML-PI" module.



CHAPTER 92

EEM Context Library Command Extensions

All the Tcl context library command extensions belong to the `::cisco::eem` namespace.

- [context_retrieve](#), on page 1925
- [context_save](#), on page 1928

context_retrieve

Retrieves Tcl variable(s) identified by the given context name, and possibly the scalar variable name, the array variable name, and the array index. Retrieved information is automatically deleted.



Note Once saved information is retrieved, it is automatically deleted. If that information is needed by another policy, the policy that retrieves it (using the **context_retrieve** command extension) should also save it again (using the **context_save** command extension).

Syntax

```
context_retrieve ctxt [var] [index_if_array]
```

Arguments

ctxt	(Mandatory) Context name.
var	(Optional) Scalar variable name or array variable name. Defaults to a null string if this argument is not specified.
index_if_array	(Optional) The array index.



Note The `index_if_array` argument will be ignored when the `var` argument is a scalar variable.

If `var` is unspecified, retrieves the whole variable table saved in the context.

If `var` is specified and `index_if_array` is not specified, or if `index_if_array` is specified but `var` is a scalar variable, retrieves the value of `var`.

If `var` is specified, and `index_if_array` is specified, and `var` is an array variable, retrieves the value of the specified array element.

Result String

Resets the Tcl global variables to the state that they were in when the save was performed.

Set_cerrno

- A string displaying `_cerrno`, `_cerr_sub_num`, `_cerr_sub_err`, `_cerr_posix_err`, `_cerr_str` due to `appl_reqinfo` error.
- Variable is not in the context.

Sample Usage

The following examples show how to use the `context_save` and `context_retrieve` command extension functionality to save and retrieve data. The examples are shown in save and retrieve pairs.

Example 1: Save

If `var` is unspecified or if a pattern is specified, saves multiple variables to the context.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvara 123
set testvarb 345
set testvarc 789
if {[catch {context_save TESTCTX "testvar*"} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}
```

Example 1: Retrieve

If `var` is unspecified, retrieves multiple variables from the context.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

if {[catch {foreach {var value} [context_retrieve TESTCTX] {set $var $value}} errmsg]} {
    action_syslog msg "context_retrieve failed: $errmsg"
} else {
    action_syslog msg "context_retrieve succeeded"
}

if {[info exists testvara]} {
    action_syslog msg "testvara exists and is $testvara"
} else {
    action_syslog msg "testvara does not exist"
}

if {[info exists testvarb]} {
    action_syslog msg "testvarb exists and is $testvarb"
}
```

```

} else {
    action_syslog msg "testvarb does not exist"
}
if {[info exists testvarc]} {
    action_syslog msg "testvarc exists and is $testvarc"
} else {
    action_syslog msg "testvarc does not exist"
}

```

Example 2: Save

If var is specified, saves the value of var.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvar 123
if {[catch {context_save TESTCTX testvar} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}

```

Example 2: Retrieve

If var is specified and index_if_array is not specified, or if index_if_array is specified but var is a scalar variable, retrieves the value of var.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
    action_syslog msg "context_retrieve failed: $errmsg"
} else {
    action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
    action_syslog msg "testvar exists and is $testvar"
} else {
    action_syslog msg "testvar does not exist"
}

```

Example 3: Save

If var is specified, saves the value of var even if it is an array.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}

```

Example 3: Retrieve

If `var` is specified, and `index_if_array` is not specified, and `var` is an array variable, retrieves the entire array.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {array set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
    action_syslog msg "context_retrieve failed: $errmsg"
} else {
    action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
    action_syslog msg "testvar exists and is [array get testvar]"
} else {
    action_syslog msg "testvar does not exist"
}
```

Example 4: Save

If `var` is specified, saves the value of `var` even if it is an array.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}
```

Example 4: Retrieve

If `var` is specified, and `index_if_array` is specified, and `var` is an array variable, retrieves the specified array element value.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar testvar1]} errmsg]} {
    action_syslog msg "context_retrieve failed: $errmsg"
} else {
    action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
    action_syslog msg "testvar exists and is $testvar"
} else {
    action_syslog msg "testvar doesn't exist"
}
```

context_save

Saves Tcl variables that match a given pattern in current and global namespaces with the given context name as identification. Use this Tcl command extension to save information outside of a policy. Saved information can be retrieved by a different policy using the **context_retrieve** command extension.



Note Once saved information is retrieved, it is automatically deleted. If that information is needed by another policy, the policy that retrieves it (using the **context_retrieve** command extension) should also save it again (using the **context_save** command extension).

Syntax

```
context_save ctxt [pattern]
```

Arguments

ctxt	(Mandatory) Context name.
pattern	(Optional) The glob-style pattern as used by the string match Tcl command. If this argument is not specified, the pattern defaults to the wildcard *. There are three constructs used in glob patterns: <ul style="list-style-type: none"> • * = all characters • ? = 1 character • [abc] = match one of a set of characters

Result String

None

Set _cerrno

A string displaying _cerrno, _cerr_sub_num, _cerr_sub_err, _cerr_posix_err, _cerr_str due to appl_setinfo error.

Sample Usage

The following examples show how to use the **context_save** and **context_retrieve** command extension functionality to save and retrieve data. The examples are shown in save and retrieve pairs.

Example 1: Save

If var is unspecified or if a pattern if specified, saves multiple variables to the context.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvara 123
set testvarb 345
set testvarc 789
if {[catch {context_save TESTCTX "testvar*"} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}
```

Example 1: Retrieve

If var is unspecified, retrieves multiple variables from the context.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

if {[catch {foreach {var value} [context_retrieve TESTCTX] {set $var $value}} errmsg]} {
    action_syslog msg "context_retrieve failed: $errmsg"
} else {
    action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvara]} {
    action_syslog msg "testvara exists and is $testvara"
} else {
    action_syslog msg "testvara does not exist"
}
if {[info exists testvarb]} {
    action_syslog msg "testvarb exists and is $testvarb"
} else {
    action_syslog msg "testvarb does not exist"
}
if {[info exists testvarc]} {
    action_syslog msg "testvarc exists and is $testvarc"
} else {
    action_syslog msg "testvarc does not exist"
}
}

```

Example 2: Save

If var is specified, saves the value of var.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvar 123
if {[catch {context_save TESTCTX testvar} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}
}

```

Example 2: Retrieve

If var is specified and index_if_array is not specified, or if index_if_array is specified but var is a scalar variable, retrieves the value of var.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
    action_syslog msg "context_retrieve failed: $errmsg"
} else {
    action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
    action_syslog msg "testvar exists and is $testvar"
} else {
}
}

```

```

        action_syslog msg "testvar does not exist"
    }

```

Example 3: Save

If var is specified, saves the value of var even if it is an array.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}

```

Example 3: Retrieve

If var is specified, and index_if_array is not specified, and var is an array variable, retrieves the entire array.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {array set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
    action_syslog msg "context_retrieve failed: $errmsg"
} else {
    action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
    action_syslog msg "testvar exists and is [array get testvar]"
} else {
    action_syslog msg "testvar does not exist"
}

```

Example 4: Save

If var is specified, saves the value of var even if it is an array.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
    action_syslog msg "context_save failed: $errmsg"
} else {
    action_syslog msg "context_save succeeded"
}

```

Example 4: Retrieve

If var is specified, and index_if_array is specified, and var is an array variable, retrieves the specified array element value.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar testvar1]} errmsg]} {

```

```
        action_syslog msg "context_retrieve failed: $errmsg"
    } else {
        action_syslog msg "context_retrieve succeeded"
    }
    if {[info exists testvar]} {
        action_syslog msg "testvar exists and is $testvar"
    } else {
        action_syslog msg "testvar doesn't exist"
    }
}
```



CHAPTER 93

EEM Event Registration Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



Note For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.



Note Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

- [event_register_appl](#), on page 1934
- [event_register_cli](#), on page 1936
- [event_register_counter](#), on page 1939
- [event_register_gold](#), on page 1941
- [event_register_identity](#), on page 1947
- [event_register_interface](#), on page 1949
- [event_register_ioswdsysmon](#), on page 1954
- [event_register_ipsla](#), on page 1957
- [event_register_mat](#), on page 1960
- [event_register_neighbor_discovery](#), on page 1962
- [event_register_nf](#), on page 1965
- [event_register_none](#), on page 1968
- [event_register_oir](#), on page 1970
- [event_register_process](#), on page 1972
- [event_register_resource](#), on page 1974
- [event_register_rf](#), on page 1976

- [event_register_routing](#), on page 1979
- [event_register_rpc](#), on page 1981
- [event_register_snmp](#), on page 1983
- [event_register_snmp_notification](#), on page 1987
- [event_register_snmp_object](#), on page 1989
- [event_register_syslog](#), on page 1992
- [event_register_timer](#), on page 1994
- [event_register_timer_subscriber](#), on page 1998
- [event_register_track](#), on page 2000
- [event_register_wdsysmon](#), on page 2002

event_register_appl

Registers for an application event. Use this Tcl command extension to run a policy when an application event is triggered following another policy's execution of an **event_publish** Tcl command extension; the **event_publish** command extension publishes an application event.

In order to register for an application event, a subsystem must be specified. Either a Tcl policy or the internal Embedded Event Manager (EEM) API can publish an application event. If the event is being published by a policy, the sub_system argument that is reserved for a policy is 798.

Syntax

```
event_register_appl [tag ?] sub_system ? type ? [queue_priority low|normal|high|last] [maxrun
?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
sub_system	(Mandatory) Number assigned to the EEM policy that published the application event. The number is set to 798 because all other numbers are reserved for Cisco use. If this argument is not specified, all components are matched.
type	(Mandatory) Event subtype within the specified event. The sub_system and type arguments uniquely identify an application event. If this argument is not specified, all types are matched. If you specify this argument, you must choose an integer between 1 and 4294967295, inclusive. There must be a match of component and type between the event_publish command extension and the event_register_appl command extension in order for the publishing and registration to work.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

If multiple conditions exist, the application event will be raised when all the conditions are satisfied.

Result String

None

Set_cerrno

No

Event_reqinfo

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"sub_system 0x%x type %u data1 {%s} data2 {%s} data3 {%s} data4 {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.

Event Type	Description
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the Embedded Event Manager (EEM).
sub_system	Number assigned to the EEM policy that published the application event. Number is set to 798 because all other numbers are reserved for Cisco use.
type	Event subtype within the specified component.
data1 data2 data3 data4	Argument data that is passed to the application-specific event when the event is published. The data is character text, an environment variable, or a combination of the two.

event_register_cli

Registers for a CLI event. Use this Tcl command extension to run a policy when a CLI command of a specific pattern is entered based on pattern matching performed against an expanded CLI command.



Note The user can enter an abbreviated CLI command, such as **sh mem summary**, and the parser will expand the command to **show memory summary** to perform the matching.



Note The functionality provided in the CLI event detector only allows a regular expression pattern match on a valid IOS CLI command itself. This does not include text after a pipe character when redirection is used.

Syntax

```
event_register_cli [tag ?] sync yes|no skip yes|no
[occurs ?] [period ?] pattern ? [default ?] [enter] [questionmark] [tab] [mode]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
sync	(Mandatory) A "yes" means that the policy (the event publish) will run synchronously with the CLI command; a "no" means that the event publish will be performed asynchronously with the CLI command. The event detector will be notified when the policy completes running. The exit status of the policy indicates whether or not the CLI command should be executed: if the exit status is zero, which means that the policy is executed successfully, the CLI command will not be executed; otherwise, the CLI command will be executed.

skip	<p>Mandatory if the sync argument is "no" and should not exist if the sync argument is "yes." If the skip argument is "yes," it means that the CLI command should not be executed. If the skip argument is "no," it means that the CLI command should be executed.</p> <p>Caution When the skip argument is "yes," unintended results may be produced if the pattern match is made for configuration commands because the CLI command that matches the regular expression will not be executed.</p>
occurs	<p>(Optional) The number of occurrences before the event is raised. If this argument is not specified, the event is raised on the first occurrence. If this argument is specified, it must be an integer between 1 and 4294967295, inclusive.</p>
period	<p>(Optional) Specifies a backward looking time window in which all CLI events must occur (the occurs clause must be satisfied) in order for an event to be published (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent event is used.</p>
pattern	<p>(Mandatory) Specifies the regular expression used to perform the CLI command pattern match.</p>
default	<p>(Optional) The time period during which the CLI event detector waits for the policy to exit (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to run the command. If this argument is not specified, the default time period is set to 30 seconds.</p>
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
enter	<p>(Optional) Specifies to perform the event match when the user presses the Enter key. When this parameter is used, the input string will not be expanded before matching.</p>
questionmark	<p>(Optional) Specifies to perform the event match when the user presses the ? key. When this parameter is used, the input string will not be expanded before matching.</p>

tab	(Optional) Specifies to perform the event match when the user presses the Tab key. When this parameter is used, the input string will not be expanded before matching.
mode	(Optional) Events will only be generated when the parser is in the specified parser mode. The available modes can be listed using the show parser dump CLI command. The mode parameter is checked when any one of the optional parameters--enter, questionmark, or tab--is specified.
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

If multiple conditions are specified, the CLI event will be raised when all the conditions are matched.

Result String

None

Set _cerno

No



Note This policy runs before the CLI command is executed. For example, suppose policy_CLI is registered to run when the **copy** command is entered. When the **copy** command is entered, the CLI event detector finds a pattern match and triggers this policy to run. When the policy execution ends, the CLI event detector determines if the **copy** command needs to be executed according to "sync", "skip" (set in the policy), and the exit status of the policy execution if needed.

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u msg {%s} msg_count %d line %u key %u tty %u error_code %u"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, at which the event was published to the EEM.
event_severity	The severity of the event.

Event Type	Description
msg	Text entered at the CLI prompt.
msg_count	Number of times the pattern matched before the event was triggered.
line	The text the parser was able to expand up to the point where the matched key was entered.
key	The enter, questionmark, or tab key.
tty	Corresponds to the line number the user is executing the command on.
error_code	The error code in CLI. 0 --No error from parser up to point where a key was entered. 1--Command is ambiguous up to point where a key was entered. 4--Unknown command up to point where a key was entered.

event_register_counter

Registers for a counter event as both a publisher and a subscriber. Use this Tcl command extension to run a policy on the basis of a named counter crossing a threshold. This event counter, as a subscriber, identifies the name of the counter to which it wants to subscribe and depends on another policy or another process to actually manipulate the counter. For example, let policyB act as a counter policy, whereas policyA (although it does not need to be a counter policy) uses **register_counter**, **counter_modify**, or **unregister_counter** Tcl command extensions to manipulate the counter defined in policyB.

Syntax

```
event_register_counter [tag ?] name ? entry_op gt|ge|eq|ne|lt|le entry_val ?
exit_op gt|ge|eq|ne|lt|le exit_val ? [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
name	(Mandatory) Name of the counter.
entry_op	(Mandatory) Entry comparison operator used to compare the current counter value with the entry value; if true, an event will be raised and event monitoring will be disabled until exit criteria are met.
entry_val	(Mandatory) Value with which the current counter value should be compared to decide if the counter event should be raised.
exit_op	(Mandatory) Exit comparison operator used to compare the current counter value with the exit value; if true, event monitoring for this event will be reenabled.

exit_val	(Mandatory) Value with which the current counter value should be compared to decide if the exit criteria are met.
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"name {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.

Event Type	Description
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
name	Counter name.

event_register_gold

Registers for a Generic Online Diagnostic (GOLD) failure event. Use this Tcl command extension to run a policy on the basis of a Generic Online Diagnostic (GOLD) failure event for the specified card and subcard.

Syntax

```
event_register_gold card all|card_number
[subcard all|subcard_number]
[new_failure TRUE|FALSE]
[severity_major TRUE]
[severity_minor TRUE]
[severity_normal TRUE]
[action_notify TRUE|FALSE]
[testing_type [bootup|ondemand|schedule|monitoring]]
[test_name [testname]]
[test_id [testnumber]]
[consecutive_failure consecutive_failure_number]
[platform_action [action_flag]]
[maxrun ?]
[queue_priority low|normal|high|last]
[nice 0|1]
```

Arguments

card	<p>(Mandatory) Specifies whether all cards or one card is to be monitored:</p> <ul style="list-style-type: none"> • card all--Specifies that all cards are to be monitored. This is the default. • card-number--Specifies that the card identified by the number card-number is to be monitored. <p>This argument must be specified to complete the event_register_goldTcl command extension.</p>
subcard	<p>(Optional) Specifies that one or more subcards are to be monitored:</p> <ul style="list-style-type: none"> • subcard all--Specifies that all subcards are to be monitored. • subcard-number--Specifies that the subcard identified by the number subcard-number is to be monitored. <p>If this argument is not specified, all subcards are monitored by default.</p>

new_failure	<p>(Optional) Specifies event criteria based on the new test failure information from GOLD:</p> <ul style="list-style-type: none"> • new_failure TRUE--Specifies that the event criterion for the new test failure is true from GOLD. • new_failure FALSE--Specifies that the event criterion for the new test failure is false from GOLD. <p>If this argument is not specified, the new test failure information from GOLD is not considered in the event criteria.</p>
severity_major	<p>(Optional) Specifies that the event criteria for diagnostic result matches with the diagnostic major error from GOLD.</p>
severity_minor	<p>(Optional) Specifies that the event criteria for diagnostic result matches with diagnostic minor error from GOLD.</p>
severity_normal	<p>(Optional) Specifies that the event criteria for diagnostic result matches with diagnostic normal from GOLD. This is the default.</p>
action_notify	<p>(Optional) Specifies the event criteria based on the action notify information from GOLD:</p> <ul style="list-style-type: none"> • action_notify TRUE--Specifies that the event criterion for the action notify is true from GOLD. • action_notify FALSE--Specifies that the event criterion for the action notify is false from GOLD. <p>If this argument is not specified, the action notify information from GOLD is not considered in the event criteria.</p>
testing_type	<p>(Optional) Specifies the event criteria based on the testing types of the diagnostic from GOLD:</p> <ul style="list-style-type: none"> • testing_type bootup--Specifies the diagnostic tests that are running on system bootup. • testing_type ondemand--Specifies the diagnostic tests that are running from CLI after the card is online. • testing_type schedule--Specifies the scheduled diagnostic tests. • testing_type monitoring--Specifies the diagnostic tests that are running periodically in the background to monitor the health of the system. <p>If this argument is not specified, the testing type information from GOLD is not considered in the event criteria and the policy applies to all the diagnostic testing types.</p>
test_name	<p>(Optional) Specifies the event criteria based on the test name:</p> <ul style="list-style-type: none"> • test_name test-name--Specifies the event criteria based on the test with the name test-name. <p>If this argument is not specified, the test name information from GOLD is not considered in the event criteria.</p>

test_id	<p>(Optional) Specifies the event criteria based on test ID:</p> <ul style="list-style-type: none"> test_id test-id--Specifies the event criteria based on the test with the ID number test-id. The maximum value of test-id is 65535. <p>Note Because the test ID can be different for the same test on different line cards, usually the test_name keyword should be used instead. If the test ID is specified and conflicts with the specified test name, the test name overwrites the test ID.</p> <p>If this argument is not specified, test ID information from GOLD is not considered in the event criteria.</p>
consecutive_failure	<p>(Optional) Specifies the event criteria based on consecutive test failure information from GOLD:</p> <ul style="list-style-type: none"> consecutive_failure consecutive-failure-number--Specifies that the event criterion is based on the occurrence of consecutive-failure-number consecutive test failures. <p>If this argument is not specified, consecutive test failure information from GOLD is not considered in the event criteria.</p>
platform_action	<p>(Optional) Specifies whether callback to the platform is needed when all the event criteria are matched. When callback is needed, the platform needs to register a callback function through the provided registry.</p> <ul style="list-style-type: none"> platform_action action-flag-number--Specifies that, when callback to the platform is needed, specific information is specified by the platform-specific action-flag-number value. The maximum value of action-flag-number is 65535. <p>Note It is up to the platform to determine what action needs to be taken based on the flag.</p> <p>If this argument is not specified, there is no callback.</p>
maxrun	<p>(Optional) Specifies the maximum runt time of the script.</p> <ul style="list-style-type: none"> maxrun max-run-time-number--Specifies that the maximum run time of the script is max-run-time-number seconds. The maximum value of max-run-time-number is 4294967295 seconds. <p>If this argument is not specified, the default run time is 20 seconds.</p>

<p>queue_priority</p>	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
<p>nice</p>	<p>(Optional) Policy run-time priority setting:</p> <ul style="list-style-type: none"> • nice 0--Specifies that the policy is run at the default run-time priority level. • nice 1--Specifies that the policy is run at a run-time priority that is less than the default priority. <p>If this argument is not specified, the default run-time priority is used.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u card %u sub_card %u"
"event_severity {%s} event_pub_sec %u event_pub_msec %u overall_result %u"
"new_failure {%s} action_notify {%s} tt %u tc %u bl %u ci %u pc %u cn {%s}"
"sn {%s} tn# {%s} ta# %s ec# {%s} rc# %u lf# {%s} tf# %u cf# %u tr# {%s}"
"tr#p# {%s} tr#d# {%s}"
```

Event Type	Description
action_notify	Action notify information in GOLD event: true or false.

Event Type	Description
bl	The boot-up diagnostic level, which can be one of the following values: <ul style="list-style-type: none"> • 0: complete diagnostic • 1: minimal diagnostics • 2: bypass diagnostic
card	Card information for the GOLD event.
cf <i>testnum</i>	Consecutive failure, where <i>testnum</i> is the test number. For example, cf3 is the EEM built-in environment variable for consecutive failure of test 3.
ci	Card index.
cn	Card name.
ec <i>testnum</i>	Test error code, where <i>testnum</i> is the test number. For example, ec3 is the EEM built-in environment variable for the error code of test 3.
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_pub_msec event_pub_sec	The time, in milliseconds and seconds, when the event was published to the EEM.
event_severity	GOLD event severity, which can be one of the following values: <ul style="list-style-type: none"> • normal • minor • major.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
lf <i>testnum</i>	Last fail time, where <i>testnum</i> is the test number. For example, lf3 is the EEM built-in variable for the last fail time of test 3. The timestamp format is <i>mmm dd yyyy hh:mm:ss</i> . For example, Mar 11 1960 08:47:00.
new_failure	The new test failure information in a GOLD event flag: true or false.
overall_result	The overall diagnostic result, which can be one of the following values: <ul style="list-style-type: none"> • 0: OK • 3: minor error • 4: major error • 14: unknown result

Event Type	Description
pc	Port counts.
rc <i>testnum</i>	Test total run count, where <i>testnum</i> is the test number. For example, rc3 is the EEM built-in variable for the total run count of test 3.
sn	Card serial number.
sub_card	The subcard on which a GOLD failure event was detected.
ta <i>testnum</i>	Test attribute, where <i>testnum</i> is the test number. For example, ta3 is the EEM built-in variable for the test attribute of test 3.
tc	Test counts.
tf <i>testnum</i>	Total failure count, where <i>testnum</i> is the test number. For example, tf3 is the EEM built-in variable for the total failure count of test 3.
tn <i>testnum</i>	Test name, where <i>testnum</i> is the test number. For example, tn3 is the EEM built-in variable for the name of test 3.
tr <i>testnum</i>	Test result, where <i>testnum</i> is the test number. For example, tr6 is the EEM built-in variable for test 6 where test 6 is not a per-port test and not a per-device test. The test result is one of the following values: <ul style="list-style-type: none"> • P: diagnostic result Pass • F: diagnostic result Fail • U: diagnostic result Unknown
tr <i>testnum</i> d <i>devnum</i>	Per-device test result, where <i>testnum</i> is the test number and <i>devnum</i> is the device number. For example, tr3d20 is the EEM built-in variable for the test result for test 3, device 20. The test result is one of the following values: <ul style="list-style-type: none"> • P: diagnostic result Pass • F: diagnostic result Fail • U: diagnostic result Unknown
tr <i>testnum</i> p <i>portnum</i>	Per-port test result, where <i>testnum</i> is the test number and <i>portnum</i> is the device number. For example, tr5p20 is the EEM built-in variable for the test result for test 3, port 20. The test result is one of the following values: <ul style="list-style-type: none"> • P: diagnostic result Pass • F: diagnostic result Fail • U: diagnostic result Unknown

Event Type	Description
tt	The testing type, which can be one of the following: <ul style="list-style-type: none"> • 1: A boot-up diagnostic • 2: An on-demand diagnostic • 3: A schedule diagnostic • 4: A monitoring diagnostic

event_register_identity

Registers for an identity event. Use this Tcl command extension to generate an event when AAA authentication or authorization is successful or failure or after normal user traffic on the port is allowed to flow.

Syntax

```
event_register_identity [tag ?] interface ?
[aaa-attribute ?]
[authc {all | fail | success}]
[authz {all | fail | success}]
[authz-complete]
[mac-address ?]
[queue_priority {normal | low | high | last}]
[maxrun ?] [nice {0 | 1}]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
interface	A regular expression pattern to match against interface names.
aaa-attribute	(Optional) A regular expression that can be used to filter events by specific AAA attributes.
authc	(Optional) Triggers events on successful, failed or both successful and failed authentication.
authz	(Optional) Triggers events on successful, failed or both successful and failed authorization.
authz-complete	(Optional) Triggers events once the device connected to the interface is fully authenticated, authorized and normal traffic has begun to flow on that interface.
mac-address	(Optional) A regular expression pattern that can be used to filter events by mac addresses of the remote device.
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo For EEM_EVENT_IDENTITY

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u identity_stage %u identity_status %u interface %u identity_mac %u
identity_<attribute> {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, at which the event was published to the EEM.
event_severity	The severity of the event.

Event Type	Description
identity_stage	One among authentication, authorization or authorization-complete stages.
identity_status	Success or one of these failure types: fail_authc, fail_aaa_server, fail_no_response, fail_timeout, fail_authz. For authorization-complete it is always success.
interface	The interface for the event.
identity_mac	The MAC address of the remote device for the event.
identity_<attribute>	For each AAA attribute, a set a dynamic variable to the value corresponding to that AAA attribute in the attribute or value list.

event_register_interface

Registers for an interface counter event. Use this Tcl command extension to generate an event when specified interface counters exceed specified thresholds.

Syntax

```
event_register_interface [tag ?] name ?
parameter ? entry_op gt|ge|eq|ne|lt|le
entry_val ? entry_val_is_increment TRUE|FALSE
entry_type value|increment|rate
[exit_comb or|and]
[exit_op gt|ge|eq|ne|lt|le]
[exit_val ?] [exit_val_is_increment TRUE|FALSE]
[exit_type value|increment|rate]
[exit_time ?] [poll_interval ?]
[average_factor ?] [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
name	(Mandatory) The name of the interface being monitored, for example, Ethernet 0/0. Abbreviations and spaces are not allowed.

parameter	<p>(Mandatory) The name of the counter being compared as follows:</p> <ul style="list-style-type: none"> • <code>input_errors</code>--Includes runs, giants, no buffer, CRC, frame, overrun, and ignored counts. Other input-related errors can also cause the input errors count to be increased, and some datagrams may have more than one error; therefore, this sum may not balance with the sum of enumerated input error counts. • <code>input_errors_crc</code>--Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. • <code>input_errors_frame</code>--Number of packets received incorrectly having a CRC error and a noninteger number of octets. • <code>input_errors_overrun</code>--Number of times the receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data. • <code>input_packets_dropped</code>--Number of packets dropped because of a full input queue. • <code>interface_resets</code>--Number of times that an interface has been completely reset. • <code>output_buffer_failures</code>--Number of failed buffers and number of buffers swapped out. • <code>output_buffer_swappedout</code>--Number of packets swapped to DRAM.
-----------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

parameter (continued)	<ul style="list-style-type: none"> • output_errors--Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, because some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories. • output_errors_underrun--Number of times that the transmitter has been running faster than the device can handle. • output_packets_dropped--Number of packets dropped because of a full output queue. • receive_broadcasts--Number of broadcast or multicast packets received by the interface. • receive_giants--Number of packets that are discarded because they exceed the maximum packet size of the medium. • receive_rate_bps--Interface receive rate in bytes per second. • receive_rate_pps--Interface receive rate in packets per second. • receive_runts--Number of packets that are discarded because they are smaller than the minimum packet size of the medium. • receive_throttle--Number of times that the receiver on the port was disabled, possibly because of buffer or processor overload. • reliability--Reliability of the interface as a fraction of 255 (255/255 is 100 percent reliability), calculated as an exponential average over 5 minutes. • rxload--Receive rate of the interface as a fraction of 255 (255/255 is 100 percent). • transmit_rate_bps--Interface transmit rate in bytes per second. • transmit_rate_pps--Interface transmit rate in packets per second. • txload--Transmit rate of the interface as a fraction of 255 (255/255 is 100 percent).
entry_op	(Mandatory) The comparison operator used to compare the current interface value with the entry value; if true, an event will be raised and event monitoring will be disabled until exit criteria are met.
entry_val	(Mandatory) The value at which the event will be triggered.
entry_val_is_increment	<p>(Mandatory) If TRUE, the entry_val field is treated as an incremental difference and is compared with the difference between the current counter value and the value when the event was last true (the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing. If FALSE, the entry_val field is compared against the current counter value.</p> <p>Note This keyword has been deprecated, and if specified, the syntax is converted into equivalent entry-type keyword syntax.</p>

entry-type	<p>Specifies a type of operation to be applied to the object ID specified by the entry-val argument.</p> <p>Value is defined as the actual value of the entry-val argument.</p> <p>Increment uses the entry-val field as an incremental difference and the entry-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.</p> <p>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change.</p>
exit_comb	<p>(Optional) Used to indicate the combination of exit condition tests required to rearm the event trigger; if the and operator is specified, both exit value and exit time tests must be true to cause rearm; if the or operator is specified, either exit value or exit time tests can be true to cause event monitoring to be rearmed.</p>
exit_op	<p>(Optional) The comparison operator used to compare the current interface value with the exit value; if true, event monitoring for this event will be reenabled.</p>
exit_val	<p>(Optional) The value at which the event is rearmed to be monitored again.</p>
exit_val_is_increment	<p>(Optional) If TRUE, the exit_val field is treated as an incremental difference and is compared with the difference between the current counter value and the value when the event was last true. A negative value checks the incremental difference for a counter that is decreasing. If FALSE, the exit_val field is compared against the current counter value.</p> <p>Note In Cisco IOS Release 12.4(20)T, this keyword is deprecated, and if specified, the syntax is converted into equivalent exit-type keyword syntax.</p>
exit-type	<p>(Optional) Specifies a type of operation to be applied to the object ID specified by the exit-val argument. If not specified, the value is assumed.</p> <p>Value is defined as the actual value of the exit-val argument.</p> <p>Increment uses the exit-val field as an incremental difference and the exit-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.</p> <p>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change.</p>

exit_time	(Optional) The time period at which the event is rearmed to be monitored again (specified in SSSSSSSSSS[.MMM] format, where SSSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999).
poll_interval	(Optional) The frequency used to collect the samples (specified in SSSSSSSSSS[.MMM] format, where SSSSSSSSSS must be an integer representing seconds between 60 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). The poll interval value must not be less than 1 second. The default is 1 second.
average-factor	(Optional) Number in the range from 1 to 64 used to calculate the period used for rate-based calculations. The average-factor value is multiplied by the poll-interval value to derive the period in milliseconds. The minimum average factor value is 1.
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSSS[.MMM] format, where SSSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Result String

None

Set _cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event_severity {%s} name {%s} parameter {%s} value %d"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event_severity	Interface event severity, which can be one of the following values: <ul style="list-style-type: none"> • normal • minor • major
name	Name of the interface.
parameter	Name of the parameter.
value	The incremental/decremental difference compared to the last event triggered or the absolute value of the parameter being monitored, depending on the specified value of entry_val_is_increment.

event_register_ioswdsysmon

Registers for an IOSWDSysMon event. Use this Tcl command extension to generate an event when a Cisco IOS task exceeds specific CPU utilization or memory thresholds. A Cisco IOS task is called a Cisco IOS process in native Cisco IOS.

Syntax

```
event_register_ioswdsysmon [tag ?] [timewin ?] [sub12op and|or] [sub1 ?] [sub2 ?]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
-----	-------------------------------------------------------------------------------------------------------------------------------------------------------

timewin	(Optional) Defines the time window within which all of the subevents must occur in order for an event to be generated (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999).
sub12_op	(Optional) The combination operator for comparison between subevent 1 and subevent 2.
sub1	(Optional) The subevent 1 specification.
sub2	(Optional) The subevent 2 specification.
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Subevent Syntax

```
cpu_proc path ? taskname ? op gt|ge|eq|ne|lt|le val ? [period ?]
mem_proc path ? taskname ? op gt|ge|eq|ne|lt|le val ? [is_percent TRUE|FALSE] [period ?]
```

Subevent Arguments

cpu_proc	(Mandatory) Specifies the use of a sample collection of CPU statistics.
path	(Mandatory) Software Modularity images only. The pathname of the POSIX process that contains the Cisco IOS scheduler to be monitored. For example, /sbin/cdp2.iosproc.
taskname	(Mandatory) The name of the Cisco IOS task to be monitored.

op	(Mandatory) The comparison operator used to compare the collected usage sample with the specified value; if true, an event will be raised.
val	(Mandatory) The value to be compared.
period	(Optional) The elapsed time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.
mem_proc	(Mandatory) Specifies the use of a sample collection of memory statistics.
is_percent	(Optional) Whether the specified value is a percentage.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"num_subs %u"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
num_subs	Number of subevents.

Where the subevent info string is for a CPU_UTIL subevent,

```
"{type %s procname {%s} pid %u taskname {%s} taskid %u value %u sec %ld msec %ld}"
```

Subevent Type	Description
type	Type of subevent.

Subevent Type	Description
procname	POSIX process name for this subevent.
pid	POSIX process ID for this subevent.
taskname	Cisco IOS task name for this subevent.
taskid	Cisco IOS task ID for this subevent.
value	Actual average CPU utilization over the measured interval.
sec , msec	Elapsed time period for this measured interval.

Where the subevent info string is for a MEM_UTIL subevent,

```
"(type %s procname {%s} pid %u taskname {%s} taskid %u is_percent %s value %u diff %d"
"sec %ld msec %ld)"
```

Subevent Type	Description
type	Type of subevent.
procname	POSIX process name for this subevent.
pid	POSIX process ID for this subevent.
taskname	Cisco IOS task name for this subevent.
taskid	Cisco IOS task ID for this subevent.
is_percent	TRUE or FALSE depending on whether the value is a percentage value.
value	Total memory use in KB or the actual average memory utilization for this measured interval.
diff	The percentage difference between the oldest sample in the measured interval and the latest sample; a negative value represents a decrease.
sec , msec	Elapsed time period for this measured interval.

event_register_ipsla

Registers for an event that is triggered by the **event ipsla** command. Use this Tcl command to publish an event when an IPSLA reaction is triggered. The group ID or the operation ID is required to register the event.

Syntax

```
event_register_ipsla [tag ?] group_name ? operation_id ? [reaction_type ?]
[dest_ip_addr ?][queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
group_name	(Mandatory) Specifies the IP SLAs group name.
operation_id	(Mandatory) Specifies the IP SLA operation ID. Number must be in the range from 1 to 2147483647.
reaction_type	<p>(Optional) Specifies the reaction to be taken for the specified IP SLAs operation.</p> <p>Type of IP SLAs reaction--One of the following keywords can be specified: connectionLoss, icpif, jitterAvg, jitterDSAvg, jitterSDAvg, maxOfNegativeDS, maxOfNegativeSD, maxOfPositiveDS, maxOfPositiveSD, mos, packetLateArrival, packetLossDS, packetLossSD, packetMIA, packetOutOfSequence, rtt, timeout or verifyError can be specified.</p> <p>Type of IP SLAs reaction. One of the following keywords can be specified:</p> <ul style="list-style-type: none"> • connectionLoss • icpif • jitterAvg • jitterDSAvg • jitterSDAvg • maxOfNegativeDS • maxOfNegativeSD • maxOfPositiveDS • maxOfPositiveSD • mos • packetLateArrival • packetLossDS • packetLossSD • packetMIA • packetOutOfSequence • rtt • timeout • verifyError
dest_ip_address	(Optional) Specifies the destination IP address of the destination port for which the IP SLAs events are monitored.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

"event_ID %u event_type %u event_pub_sec %u event_pub_msec %u event_severity %u" "group_name %u operation_id %u condition %u reaction_type %u dest_ip_addr %u" "threshold_rising %u threshold_falling %u measured_threshold_value %u" "threshold_count1 %u threshold count2 %u"

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	The type of event to monitor for the create, update, and delete flow.
event_type_string	An ASCII string that represents the name of the event for this event type.

Event Type	Description
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event_severity	The severity of the event.
group_name	The name of the IPSLA group.
operation_id	The IPSLA operation ID.
condition	The condition of IPSLA, which can be one of the following: <ul style="list-style-type: none"> cleared occurred
reaction_type	The IPSLA reaction type.
dest_ip_address	The IPSLA destination IP address.
threshold rising	The IPSLA configured rising threshold value.
threshold falling	The IPSLA configured falling threshold value.
measured_threshold_value	The measured threshold value of the IPSLA operation.
threshold_count1	Corresponds to the argument of the threshold type1.
threshold_count2	Corresponds to the argument of the threshold type2.

event_register_mat

Registers for a MAT event. Use this Tcl command extension to generate an event when a mac-address is learned in the mac-address-table.

Syntax

```
event_register_identity [tag ?] interface ?
[mac-address ?]
[type {add | delete}]
[hold-down ?]
[maxrun ?]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
interface	A regular expression pattern to match against interface names.
mac-address	Mandatory if the interface parameter is not specified. A regular expression pattern that can be used to filter events by mac addresses of the remote device.

type	(Optional) Filter based on a mac-address-table event type of add or delete. If not specified, the event type is not used in determining whether the event should be triggered.
hold-down	(Optional) When a mac-address-table event comes in, the hold-down timer can be set to make the event to wait between 1 and 4294967295 seconds before processing the policy. If not set then the policy is not delayed in being processed.
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSSS[.MMM] format, where SSSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.

Result String

None

Set_cerrno

No

Event_reqinfo For EEM_EVENT_MAT

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u notification %u intf_name %u mac_address {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, at which the event was published to the EEM.
event_severity	The severity of the event.
notification	Notification type--add or delete.
intf_name	The interface name for the address table entry.
mac_address	The mac-address for the address table entry.

event_register_neighbor_discovery

Registers for a neighbor discover event. Use this Tcl command extension to generate an event when a Cisco Discovery Protocol (CDP) or Link Layer Discovery Protocol (LLDP) cache entry or a interface link status changes.

Syntax

```
event_register_neighbor_discovery [tag ?] interface ?
[cdp {add | update | delete | all}]
[lldp {add | update | delete | all}]
[link-event]
[line-event]
[queue_priority {normal | low | high | last}]
[maxrun ?] [nice {0 | 1}]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
interface	A regular expression pattern to match against interface names.
cdp	Trigger an event when a matching CDP event occurs. One of the following options should be specified. <ul style="list-style-type: none"> • add--Trigger events only when a new CDP cache entry is created in the CDP table. • all--Trigger an event when a CDP cache entry is added or deleted from the CDP cache table and when a remote CDP device sends a keepalive to update the CDP cache entry. • delete--trigger events only when a CDP cache entry is deleted from the CDP table. • update--trigger an event when a CDP cache entry is added to the CDP table or when the remote CDP device sends a CDP keepalive to update the CDP cache entry.
lldp	Trigger an event when a matching lldp event occurs. One of the following options should be specified. <ul style="list-style-type: none"> • add--Trigger events only when a new cdp cache entry is created in the cdp table. • all--Trigger an event when a cdp cache entry is added or deleted from the cdp cache table and when a remote cdp device sends a keepalive to update the cdp cache entry. • delete--trigger events only when a cdp cache entry is deleted from the cdp table. • update--trigger an event when a cdp cache entry is added to the cdp table or when the remote cdp device sends a cdp keepalive to update the cdp cache entry.
line-event	Trigger an event when the interface line protocol status changes.
link-event	Trigger an event when the interface link status changes.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo For EEM_EVENT_NEIGHBOR_DISCOVERY

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u event_severity %u nd_notification {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, at which the event was published to the EEM.

Event Type	Description
event_severity	The severity of the event.
Common Event_Reqinfo	
nd_notification	The type of notification--cdp-add, cdp-update, cdp-delete, lldp-add, lldp-update, lldp-delete, link, line.
nd_intf_linkstatus	The current interface link status, up or down.
nd_intf_linestatus	The current interface line status, down, goingdown, init, testing, up, reset, admindown, deleted.
nd_local_intf_name	The local interface name for the event.
nd_short_local_intf_name	The short name of the local interface for the event.
nd_port_id	The port id as identified by either the cdp or lldp protocol. This is not set for link or line protocol events.
CDP-specific Event_reqinfo	
nd_protocol	Identifies which protocol triggered the event, for CDP it will always be set to cdp.
nd_proto_notif	Identifies which type of protocol event triggered the event, add, update or delete.
nd_proto_new_entry	If set to 1, the event was triggered because the cache entry is new, otherwise it will be set to 0.
nd_cdp_entry_name	The name of the cdp cache entry in the cdp table.
nd_cdp_hold_time	The time remaining until the cdp cache entry expires and is deleted from the cdp table. This time will be reset to some maximum by an update from the cdp neighbor. It is usually set to 0 for new entries.
nd_cdp_mgmt_domain	The CDP VTP management domain.
nd_cdp_platform	The platform name reported by the remote device.
nd_cdp_version	The version of code running on the remote device.
nd_cdp_capabilities_string	The contents of the CDP capabilities field in a string format: Router, Trans-Bridge, Source-Route-Bridge, Switch, Host, IGMP, Repeater, Phone, Remotely-Managed device, CVTA phone port, Two-port Mac Relay or any combination of these separated by commas.
nd_cdp_capabilities_bits	The CDP capabilities bits in a hexadecimal number preceded with 0x.
nd_cdp_capabilities_bit_[0-31]	A series of values that will be set to YES if that bit in the capabilities field is set or NO if it is not set.

Event Type	Description
LLDP-specific Event_reqinfo	
nd_protocol	Identifies which protocol triggered the event, for LLDP it will always be set to lldp.
nd_proto_notif	Identifies which type of protocol event triggered the event, add, update or delete.
nd_proto_new_entry	If set to 1, the event was triggered because the cache entry is new, otherwise it will be set to 0.
nd_lldp_chassis_id	The chassis id field from the LLDP cache entry.
nd_lldp_system_name	The system name from the LLDP cache entry.
nd_lldp_system_description	The system description field from the LLDP cache entry.
nd_lldp_ttl	The LLDP time to live field from the LLDP cache entry.
nd_lldp_port_description	The port description field from the LLDP cache entry.
nd_lldp_system_capabilities_string	The LLDP system capabilities field from the LLDP cache entry. Provided as a string that can contain O, P, B, W, R, T, C, S or any combination of these separated by commas.
nd_lldp_enabled_capabilities_string	The LLDP enabled system capabilities field from the LLDP cache entry. Provided as a string that can contain O, P, B, W, R, T, C, S or any combination of these separated by commas.
nd_lldp_system_capabilities_bits	The LLDP system capabilities bits field from the LLDP cache entry. Provided as a hexadecimal number preceded by 0x.
nd_lldp_enabled_capabilities_bits	The LLDP enabled capabilities bits field from the LLDP cache entry. Provided as a hexadecimal number preceded by 0x.
nd_lldp_capabilities_bits	The LLDP capabilities bits field from the LLDP cache entry. Provided as a hexadecimal number preceded by 0x.
nd_lldp_capabilities_bit_[0-31]	A series of values that will be set to YES if that bit in the capabilities field is set or NO if it is not set.

event_register_nf

Registers for an event when a NetFlow event is triggered by the **event nf** command. Use this Tel command to publish an event when an NetFlow reaction is triggered..

Syntax

```
event_register_nf [tag ?] monitor_name ? event_type create|update|delete
exit_event_type create|update|delete event1-event4 ? [maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
monitor_name	(Mandatory) The name of the NetFlow monitor.
event_type	(Mandatory) The type of event to monitor for the create, update, and delete flow.
exit_event_type	(Mandatory) The event-type (create, delete, update) at which the event is rearmed to be monitored again.
event1- event4	(Mandatory) Specifies the event and its attributes to monitor. Valid values are event1 , event2 , event3 , and event4 . The subevent keywords can be used alone, together, or in any combination with each other, but each keyword can be used only once.
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Subevent Syntax

```
field ? rate_interval ? event1 only entry_value ? entry_op eq|ge|gt|le|lt|wc
[exit_value ?] [exit_op eq|ge|gt|le|lt|wc] [exit_rate_interval ? event1 only]
```

Subevent Arguments

field	(Mandatory) Specifies the cache or field attribute to be monitored. One of the following attributes can be specified: <ul style="list-style-type: none"> • counter {bytes packets}--Specifies the counter fields. • datalink {dot1q mac}--Specifies the datalink (layer2) fields. • flow {direction sampler}--Specifies the flow identifying fields. • interface {input output}--Specifies the interface fields. • ipv4 <i>field-type</i>-- Specifies the IPv4 fields. • ipv6 <i>field-type</i>-- IPv6 fields • routing <i>routing-attribute</i> -- Specifies the routing attributes. • timestamp sysuptime {first last}--Specifies the timestamp fields. • transport <i>field-type</i>-- Specifies the Transport layer fields.
rate_interval	(Mandatory) Specifies the rate interval value in seconds used to calculate the rate. This field is only valid for event1.

entry_value	(Mandatory) Specifies the field or rate value.
entry_op	(Mandatory) Specifies the field operator. The comparison operator valid values are: <ul style="list-style-type: none"> • eq - Equal to • ge - Greater than or equal to • gt - Greater than • le - Less than or equal to • lt - Less than • wc - Wildcard
exit_value	(Optional) The value at which the event is rearmed to be monitored again.
exit_op	(Optional) The comparison operator used to compare the current event field or rate value with the exit value; if true, event monitoring for this event is reenabled. The comparison operator valid values are: <ul style="list-style-type: none"> • eq - Equal to • ge - Greater than or equal to • gt - Greater than • le - Less than or equal to • lt - Less than • wc - Wildcard
exit_rate_interval	(Optional) Specifies the exit rate interval value in seconds used to calculate the exit rate value. This field is only valid for event1.

Result String

None

Set_cerrno

No

Event_reqinfo

"event_ID %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u event_severity %u monitor_name %u event1-event4_field %u event1-event4_value"

Event Type	Description
------------	-------------

event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	The type of event to monitor for the create, update, and delete flow.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event_severity	The severity of the NetFlow event.
monitor_name	The name of the NetFlow monitor.
event1-event4_field	Specifies the event and its attributes to monitor. Valid values are event1 , event2 , event3 , and event4 .
event1-event4_value	Specifies the event value and its attributes to monitor. Valid values are event1 , event2 , event3 , and event4 .

event_register_none

Registers for an event that is triggered by the **event manager run** command. These events are handled by the None event detector that screens for this event.

Syntax

```
event_register_none [tag ?] [sync {yes|no}] [default ?] [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
sync	(Optional) A "yes" or a "no" is required to complete this keyword. <ul style="list-style-type: none"> • If the yes keyword is specified, the policy will run synchronously with the CLI command. • If the no keyword is specified, the policy will run asynchronously with the CLI command.
default	(Optional) The time period during which the CLI event detector waits for the policy to exit (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to run the command. If this argument is not specified, the default time period is set to 30 seconds.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u arg %u"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.

event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event_severity	The severity of the event.
argc arg1 arg2 arg3 arg4 arg6 arg7 arg8 arg9 arg10 arg11 arg12 arg13 arg14 arg15	The parameters that are passed from the XML SOAP command to the script.

event_register_oir

Registers for an online insertion and removal (OIR) event. Use this Tcl command extension to run a policy on the basis of an event raised when a hardware card OIR occurs. These events are handled by the OIR event detector that screens for this event.

Syntax

```
event_register_oir [tag ?] [queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
-----	-------------------------------------------------------------------------------------------------------------------------------------------------------

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
 "slot %u event %s"

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event ID.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.

Event Type	Description
slot	Slot number for the affected card.
event	Indicates a string, removed or online, that represents either an OIR removal event or an OIR insertion event.

event_register_process

Registers for a process event. Use this Tcl command extension to run a policy on the basis of an event raised when a Cisco IOS Software Modularity process starts or stops. These events are handled by the System Manager event detector that screens for this event. This Tcl command extension is supported only in Software Modularity images.

Syntax

```
event_register_process [tag ?] abort|term|start|user_restart|user_shutdown
[sub_system ?] [version ?] [instance ?] [path ?] [node ?]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
abort	(Mandatory) Abnormal process termination. Process may terminate because of exiting with a nonzero exit status, receiving a kernel-generated signal, or receiving a SIGTERM or SIGKILL signal that is not sent because of user request.
term	(Mandatory) Normal process termination.
start	(Mandatory) Process start.
user_restart	(Mandatory) Process termination due to the process restart request from the CLI command.
user_shutdown	(Mandatory) Process termination due to the process kill request from the CLI command.
sub_system	(Optional) Number assigned to the EEM policy that published the process event. Number is set to 798 because all other numbers are reserved for Cisco use.
version	(Optional) Version number of the process assigned by the version manager. Must be of the form major_number.minor_number.level. If specified, each component of the version number must be an integer between 1 and 4294967295, inclusive.
instance	(Optional) Process instance ID. If specified, this argument must be an integer between 1 and 4294967295, inclusive.
path	(Optional) Process pathname (a regular expression string). If the value of the process-name argument contains embedded blanks, enclose it in double quotation marks. Use path ".*" to match all processes.

node	<p>(Optional) The node name is a string that consists of the word "node" followed by two fields separated by a slash character using the following format:</p> <p>node<slot-number>/<cpu-number></p> <p>The slot-number is the hardware slot number. The cpu-number is the hardware CPU number. For example, the SP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be specified as node0/0. The RP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be addressed as node0/1. If the node argument is not specified, the default node specification is always the regular expression pattern match of * representing all applicable nodes.</p>
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

If an optional argument is not specified, the event matches all possible values of the argument. If multiple arguments are specified, the process event will be raised when all the conditions are matched.

Result String

None

Set_cerrno

No

Event_reqinfo

"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"

```
"sub_system 0x%x instance %u process_name {%s} path {%s} exit_status 0x%x"
"respawn_count %u last_respawn_sec %ld last_respawn_msec %ld fail_count %u"
"dump_count %u node_name {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
sub_system	Number assigned to the EEM policy that published the application-specific event. Number is set to 798 because all other numbers are reserved for Cisco use.
instance	Process instance ID.
process_name	Process name.
path	Process absolute name including path.
exit_status	Process last exit status.
respawn_count	Number of times that the process was restarted.
last_respawn_sec last_respawn_msec	The calendar time when the last restart occurred.
fail_count	Number of restart attempts of the process that failed. This count will be reset to 0 when the process is successfully restarted.
dump_count	Number of core dumps taken of the process.
node_name	Name of the node that the process is on. The node name is a string that consists of the word "node" followed by two fields separated by a slash character using the following format: node <i>slot-number / cpu-number</i> The slot-number is the hardware slot number. The cpu-number is the hardware CPU number.

event_register_resource

Registers for an Embedded Resource Manager (ERM) event. Use this Tcl command extension to run a policy on the basis of an ERM event report for a specified policy. ERM events are screened by the EEM Resource event detector, allowing an EEM policy to be run when a match occurs for the specified ERM policy.

Syntax

```
event_register_resource policy policy-name [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

Arguments

policy	(Mandatory) Specifies the use of a policy.
policy-name	(Mandatory) Name of an ERM policy.
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"owner_id %lld user_id %lld" time_sent %llu dampen_time %d notify_data_flags %u"
"level {%s} direction {%s} configured_threshold %u current_value %u"
"policy_violation_flag {%s} policy_id %d"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
owner_id	The Embedded Resource Manager (ERM) owner ID.
user_id	The ERM user ID.
time_sent	The ERM event time, in nanoseconds.
dampen_time	The ERM dampen time, in nanoseconds.
notify_data_flags	The ERM notify data flag.
level	The ERM event level. The four event levels are normal, minor, major, and critical.
direction	The ERM event direction. The event direction can be one of the following: up, down, or no change.
configured_threshold	The configured ERM threshold.
current_value	The current value reported by ERM.
policy_violation_flag	The ERM policy violation flag; either false or true.
policy_id	The ERM policy ID.

event_register_rf

Registers for a Redundancy Facility (RF) event. Use this Tcl command extension to run a policy when an RF progression or status event notification occurs.

Syntax

```
event_register_rf [tag ?] event ?
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```


Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
event	<p>(Mandatory) Name of the RF progression or status event. Valid values are:</p> <ul style="list-style-type: none"> • RF_PROG_ACTIVE • RF_PROG_ACTIVE_DRAIN • RF_PROG_ACTIVE_FAST = 200 • RF_PROG_ACTIVE_PRECONFIG • RF_PROG_ACTIVE_POSTCONFIG • RF_PROG_EXTRALOAD • RF_PROG_HANDBACK • RF_PROG_INITIALIZATION • RF_PROG_PLATFORM_SYNC • RF_PROG_STANDBY_BULK • RF_PROG_STANDBY_COLD • RF_PROG_STANDBY_CONFIG • RF_PROG_STANDBY_FILESYS • RF_PROG_STANDBY_HOT • RF_PROG_STANDBY_OIR_SYNC_DONE • RF_REGISTRATION_STATUS • RF_STATUS_MAINTENANCE_ENABLE • RF_STATUS_MANUAL_SWACT • RF_STATUS_OPER_REDUNDANCY_MODE_CHANGE • RF_STATUS_PEER_COMM • RF_STATUS_PEER_PRESENCE • RF_STATUS_REDUNDANCY_MODE_CHANGE • RF_STATUS_SWACT_INHIBIT

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.

Event Type	Description
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event	RF progression or status event notification that caused this event to be published.

event_register_routing

Registers for an event that is triggered by the **event routing** command. These events are handled by the routing event detector to publish an event when route entries change in Routing Information Base (RIB) infrastructure. Use this Tcl command extension to run a routing policy for this script. The network IP address for the route to be monitored must be specified.

Syntax

```
event_register_routing [tag ?] network ? length [ge|le|ne] [type add|remove|modify|all]
[protocol ?] [queue_priority normal|low|high|last] [maxrun ?] [nice {0 | 1}]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
network	Specifies the network IP address. The network number can be any valid IP address or prefix.
length	Specifies the length of the network mask in bits. The bit mask can be a number from 0 to 32. <ul style="list-style-type: none"> • ge --(Optional) Specifies the minimum prefix length to be matched. The ge keyword represents greater than or equal to operator. • le --(Optional) Specifies the maximum prefix length to be matched. The le keyword represents the less than or equal to operator. • ne --(Optional) Specifies the prefix length not to be matched. The ne keyword represents not equal to operator. <p>When ge, le and ne keywords are not configured, an exact match of network length is processed.</p>
type	(Optional) Specifies the desired policy trigger. The type options are add , remove , modify , and all . The default is all .
protocol	(Optional) Specifies the protocol value for the network being monitored. One of the following protocols can be used: all , bgp , connected , eigrp , isis , iso-igrp , mobile , odr , ospf , rip , and static . The default is all .

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
 "event_severity {%s} %u network %u mask %u protocol %u lastgateway %u distance %u" "time_sec %u
 time_msec %u metric %u lastinterface %u"

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.

Event Type	Description
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event_severity	The severity of the event.
network	The network prefix in IP address format
mask	The network mask in IP address format
protocol	Type of network protocol.
type	Type of event to add, remove or modify.
lastgateway	The last known gateway.
distance	The administrative distance.
time_sec time_msec	Time of event in seconds and milliseconds, when the event was published to the EEM.
metric	Path metric.
lastinterface	The last known interface.

event_register_rpc

Registers for an event that is triggered by the EEM SSH Remote Procedure Call (RPC) command. These events are handled by the RPC event detector that screens for this event. Use this Tcl command extension to run a RPC policy for this script.

Syntax

```
event_register_rpc [queue_priority {normal | low | high | last}] [maxrun <sec.msec>] [nice
{0 | 1}] [default <sec.msec>]
```

Arguments

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>
default	<p>(Optional) The time period during which the CLI event detector waits for the policy to exit (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to run the command. If this argument is not specified, the default time period is set to 30 seconds.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u arg %u"

Event Type	Description
------------	-------------

event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
argc arg0 arg1 arg2 arg3 arg4 arg6 arg7 arg8 arg9 arg10 arg11 arg12 arg13 arg14	The parameters that are passed from the XML SOAP command to the script.

event_register_snmp

Registers for a Simple Network Management Protocol (SNMP) statistics event. Use this Tcl command extension to run a policy when a given counter specified by an SNMP object ID (oid) crosses a defined threshold.

Syntax

```
event_register_snmp [tag ?] oid ? get_type exact|next
entry_op gt|ge|eq|ne|lt|le entry_val ?
entry_type value|increment|rate
[exit_comb or|and]
[exit_op gt|ge|eq|ne|lt|le] [exit_val ?]
[exit_type value|increment|rate]
[exit_time ?] poll_interval ? [average_factor ?]
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
oid	(Mandatory) OID number of data element in SNMP dot notation (for example, 1.3.6.1.2.1.2.1.0). The types of OIDs allowed are: <ul style="list-style-type: none"> • COUNTER_TYPE • COUNTER_64_TYPE • GAUGE_TYPE • INTEGER_TYPE • OCTET_PRIM_TYPE • OPAQUE_PRIM_TYPE • TIME_TICKS_TYPE
entry_op	(Mandatory) Entry comparison operator used to compare the current OID data value with the entry value; if true, an event will be raised and event monitoring will be disabled until exit criteria are met.
get_type	(Mandatory) Type of SNMP get operation that needs to be applied to the OID specified. If the get_type argument is "exact," the value of the specified OID is retrieved; if the get_type argument is "next," the value of the lexicographical successor to the specified OID is retrieved.
entry_val	(Mandatory) Value with which the current oid data value should be compared to decide if the SNMP event should be raised.
entry-type	Specifies a type of operation to be applied to the object ID specified by the entry-val argument. Value is defined as the actual value of the entry-val argument. <p>Increment uses the entry-val field as an incremental difference and the entry-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.</p> <p>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change.</p>
exit_comb	(Optional) Exit combination operator used to indicate the combination of exit condition tests required to decide if the exit criteria are met so that the event monitoring can be reenabled. If it is "and," both exit value and exit time tests must be passed to meet the exit criteria. If it is "or," either exit value or exit time tests can be passed to meet the exit criteria. When exit_comb is "and," exit_op, and exit_val (exit_time) must exist. When exit_comb is "or," (exit_op and exit_val) or (exit_time) must exist.
exit_op	(Optional) Exit comparison operator used to compare the current oid data value with the exit value; if true, event monitoring for this event will be reenabled.

exit_val	(Optional) Value with which the current oid data value should be compared to decide if the exit criteria are met.
exit-type	<p>(Optional) Specifies a type of operation to be applied to the object ID specified by the exit-val argument. If not specified, the value is assumed.</p> <p>Value is defined as the actual value of the exit-val argument.</p> <p>Increment uses the exit-val field as an incremental difference and the exit-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.</p> <p>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change.</p>
exit_time	(Optional) Number of POSIX timer units after an event is raised when event monitoring will be enabled again. Specified in SSSSSSSSS[.MMM] format where SSSSSSSSS must be an integer number representing seconds between 0 and 4294967295, inclusive. MMM represents milliseconds and must be an integer number between 0 and 999.
poll_interval	(Mandatory) Interval between consecutive polls in POSIX timer units. Currently the interval is forced to be at least 1 second (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999).
average-factor	(Optional) Number in the range from 1 to 64 used to calculate the period used for rate-based calculations. The average-factor value is multiplied by the poll-interval value to derive the period in milliseconds. The minimum average factor value is 1.
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>

maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event_severity {%s} oid {%s} val {%s} delta_val {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event_severity	SNMP event severity, which can be one of the following values: <ul style="list-style-type: none"> • normal • minor • major
oid	Object ID of data element, in SNMP dot notation.
val	Value of the data element.
delta_val	Delta value between the value of the policies.

event_register_snmp_notification

Registers for a Simple Network Management Protocol (SNMP) notification trap event. Use this Tcl command extension to run a policy when an SNMP trap with the specified SNMP object ID (oid) is encountered on a specific interface or address. The **snmp-server manager** CLI command must be enabled for the SNMP notifications to work using Tcl policies.

Syntax

```
event_register_snmp_notification [tag ?] oid ? oid_val ?
op {gt|ge|eq|ne|lt|le}
[maxrun ?]
[src_ip_address ?]
[dest_ip_address ?]
[queue_priority {normal|low|high|last}]
[maxrun ?]
[nice {0|1}]
[default ?]
[direction {incoming|outgoing}]
[msg_op {drop|send}]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
oid	(Mandatory) OID number of the data element in SNMP dot notation (for example, 1.3.6.1.2.1.2.1.0). If the specified OID ends with a dot (.), then all OIDs that start with the OID number before the dot are matched. The types of OIDs allowed are: <ul style="list-style-type: none"> • COUNTER_TYPE • COUNTER_64_TYPE • GAUGE_TYPE • INTEGER_TYPE • OCTET_PRIM_TYPE • OPAQUE_PRIM_TYPE • TIME_TICKS_TYPE
oid_val	(Mandatory) OID value with which the current OID data value should be compared to decide if the SNMP event should be raised.
op	(Mandatory) Comparison operator used to compare the current OID data value with the SNMP Protocol Data Unit (PDU) OID data value; if this is true, an event is raised.
maxrun	(Optional) Maximum run time of the script (specified in sssssss[.mmm] format, where sssssss must be an integer representing seconds between 0 and 31536000, inclusive, and where mmm must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.

src_ip_address	(Optional) Source IP address where the SNMP notification trap originates. The default is all; it is set to receive SNMP notification traps from all IP addresses.
dest_ip_address	(Optional) Destination IP address where the SNMP notification trap is sent. The default is all; it is set to receive SNMP traps from all destination IP addresses.
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the queue_priority_last argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
default	(Optional) Specifies the time period in seconds during which the snmp notification event detector waits for the policy to exit. The time period is specified in ssssssss[.mmm] format, where ssssssss must be an integer representing seconds between 0 and 4294967295 and mmm must be an integer representing milliseconds between 0 and 999.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.
direction	(Optional) The direction of the incoming or outgoing SNMP trap or inform PDU to filter. The default value is incoming.
msg_op	(Optional) The action to be taken on the SNMP PDU (drop it or send it) once the event is triggered. The default value is send.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u
event_severity {%s}" "oid {%s} oid_val {%s} src_ip_addr {%s} dest_ip_addr {%s} x_x_x_x_x
(varbinds) {%s} trunc_vb_buf {%s} trap_oid {%s} enterprise_oid {%s} generic_trap %u
specific_trap %u"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
oid	An user specified object ID.
oid_val	An user specified object ID value.
src_ip_addr	The source IP address of the SNMP protocol data unit (PDU).
dest_ip_addr	The destination IP address of the SNMP PDU.
x_x_x_x_x (varbinds)	The SNMP PDU varbind information.
trap_oid	Indicates the trap OID value.
enterprise_oid	Indicates the enterprise OID value.
generic_trap	Indicates one of a number of generic trap types. There are seven generic trap numbers zero to six.
specific_trap	Indicates one of a number of specific trap codes.

event_register_snmp_object

Registers for a Simple Network Management Protocol (SNMP) object event. Use this Tcl command extension to replace the value when an SNMP with the specified SNMP-object ID (OID) is encountered on a specific interface or address.

Syntax

```
event_register_snmp_object oid ?
type {int|uint|counter|counter64|gauge|ipv4||oid|string}
sync {yes|no}
skip {yes|no}
[istable {yes|no}]
[default ?]
[queue_priority {normal|low|high|last}]
[maxrun ?]
[nice {0|1}]
```

Arguments

oid	<p>(Mandatory) OID number of the data element in SNMP dot notation (for example, 1.3.6.1.2.1.2.1.0). If the specified OID ends with a dot (.), then all OIDs that start with the OID number before the dot are matched. The types of OIDs allowed are:</p> <ul style="list-style-type: none"> • COUNTER_TYPE • COUNTER_64_TYPE • GAUGE_TYPE • INTEGER_TYPE • OCTET_PRIM_TYPE • OPAQUE_PRIM_TYPE • TIME_TICKS_TYPE
type	(Mandatory) OID value type.
sync	<p>(Mandatory) A "yes" means that the EEM policy will be notified. If the applet set_exit_status or Tcl return value is 0, then SNMP will handle the request. If the return value is 1, SNMP will use the value provided by the policy for the get request and will not process the set request. A "no" means that EEM will not be notified and SNMP will handle the request.</p> <p>Only one OID can be associated with a synchronous policy. However, multiple synchronous policies can be registered for the same OID.</p>
skip	Mandatory if the sync argument is "no" and should not exist if the sync argument is "yes." If the skip argument is "yes," it means that SNMP will handle the request. If the skip argument is "no," it means that SNMP will act as if the object does not exist.
istable	(Optional) A value of "no" means the OID is scalar object, and "yes" means the OID is table object.
default	(Optional) The time period during which the SNMP Object event detector waits for the policy to exit (specified in ssssssss[.mmm] format, where ssssssss must be an integer representing seconds between 0 and 4294967295, inclusive, and where mmm must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to process the set or get request normally by SNMP subsystem. If this argument is not specified, the default time period is set to 30 seconds.
maxrun	(Optional) Maximum run time of the script (specified in ssssssss[.mmm] format, where ssssssss must be an integer representing seconds between 0 and 31536000, inclusive, and where mmm must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the queue_priority_last argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u event_severity {%s}" "oid {%s} request {%s} request_type {%s} value %u"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
event_severity	The severity of the event.
oid	The ID of the SNMP object in the received get or set request.

Event Type	Description
request	The get or set request type.
request_type	The type of request (exact or next).
value	For set requests only. The value to set the object to.

event_register_syslog

Registers for a syslog event. Use this Tcl command extension to trigger a policy when a syslog message of a specific pattern is logged after a certain number of occurrences during a certain period of time.

Syntax

```
event_register_syslog [tag ?] [occurs ?] [period ?] pattern ?
[priority all|emergencies|alerts|critical|errors|warnings|notifications|
informational|debugging|0|1|2|3|4|5|6|7]
[queue_priority low|normal|high|last]
[severity_fatal] [severity_critical] [severity_major]
[severity_minor] [severity_warning] [severity_notification]
[severity_normal] [severity_debugging]
[maxrun ?] [nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
occurs	(Optional) Number of occurrences before the event is raised; if not specified, the event is raised on the first occurrence. If specified, the value must be greater than 0.
period	(Optional) Time interval, in seconds and milliseconds, during which the one or more occurrences must take place in order to raise an event (specified in SSSSSSSSS[.MMM] format where SSSSSSSSS must be an integer number representing seconds between 0 and 4294967295, inclusive, and where MMM represents milliseconds and must be an integer number between 0 and 999). If this argument is not specified, no period check is applied.
pattern	(Mandatory) A regular expression used to perform syslog message pattern match. This argument is what the policy uses to identify the logged syslog message.
priority	(Optional) The message priority to be screened. If this argument is specified, only messages that are at the specified logging priority level, or lower, are screened. If this argument is not specified, the default priority is 0.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>
severity_xxx	<p>(Optional) The event severity to be screened. If this argument is specified, only messages that are at the specified severity level are screened. See the table titled "Severity Level Mapping For Syslog Events" for the severity level mapping for syslog events.</p>

If multiple conditions are specified, the syslog event will be raised when all the conditions are matched.

Table 202: Severity Level Mapping For Syslog Events

Severity Keyword	Syslog Priority	Description
severity_fatal	LOG_EMERG (0)	System is unusable.
severity_critical	LOG_ALERT (1)	Critical conditions, immediate attention required.
severity_major	LOG_CRIT (2)	Major conditions.
severity_minor	LOG_ERR (3)	Minor conditions.
severity_warning	LOG_WARNING (4)	Warning conditions.
severity_notification	LOG_NOTICE (5)	Basic notification, informational messages.
severity_normal	LOG_INFO (6)	Normal event, indicates returning to a normal state.
severity_debugging	LOG_DEBUG (7)	Debugging messages.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"msg {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
msg	The last syslog message that matches the pattern.

event_register_timer

Creates a timer and registers for a timer event as both a publisher and a subscriber. Use this Tel command extension when there is a need to trigger a policy that is time specific or timer based. This event timer is both an event publisher and a subscriber. The publisher part indicates the conditions under which the named timer is to go off. The subscriber part identifies the name of the timer to which the event is subscribing.



Note Both the CRON and absolute time specifications work on local time.

Syntax

```
event_register_timer [tag ?] watchdog|countdown|absolute|cron
[name ?] [cron_entry ?]
[time ?]
[queue_priority low|normal|high|last] [maxrun ?]
[nice 0|1]
```

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
watchdog	(Mandatory) Watchdog timer.
countdown	(Mandatory) Countdown timer.
absolute	(Mandatory) Absolute timer.
cron	(Mandatory) CRON timer.
name	(Optional) Name of the timer.
cron_entry	<p>(Optional) Must be specified if the CRON timer type is specified. Must not be specified if any other timer type is specified. A cron_entry is a partial UNIX crontab entry (the first five fields) as used with the UNIX CRON daemon.</p> <p>A cron_entry specification consists of a text string with five fields. The fields are separated by spaces. The fields represent the time and date when CRON timer events will be triggered. The fields are described in the table titled "Time and Date When CRON Events Will Be Triggered."</p> <p>Ranges of numbers are allowed. Ranges are two numbers separated with a hyphen. The specified range is inclusive. For example, 8-11 for an hour entry specifies execution at hours 8, 9, 10, and 11.</p> <p>A field may be an asterisk (*), which always stands for "first-last."</p> <p>Lists are allowed. A list is a set of numbers (or ranges) separated by commas. Examples: "1,2,5,9" and "0-4,8-12".</p> <p>Step values can be used in conjunction with ranges. Following a range with "/<number>" specifies skips of the number's value through the range. For example, "0-23/2" can be used in the hour field to specify an event that is triggered every other hour. Steps are also permitted after an asterisk, so if you want to say "every two hours", use "* /2".</p> <p>Names can also be used for the month and the day of week fields. Use the first three letters of the particular day or month (case does not matter). Ranges or lists of names are not allowed.</p> <p>The day on which a timer event is triggered can be specified by two fields: day of month and day of week. If both fields are restricted (that is, are not *), an event will be triggered when either field matches the current time. For example, "30 4 1,15 * 5" would cause an event to be triggered at 4:30 a.m. on the 1st and 15th of each month, plus every Friday.</p> <p>Instead of the first five fields, one of seven special strings may appear. These seven special strings are described in the table titled "Special Strings for cron_entry."</p> <p>Example 1: "0 0 1,15 * 1" would trigger an event at midnight on the 1st and 15th of each month, as well as on every Monday. To specify days by only one field, the other field should be set to *; "0 0 * * 1" would trigger an event at midnight only on Mondays.</p> <p>Example 2: "15 16 1 * *" would trigger an event at 4:15 p.m. on the first day of each month.</p> <p>Example 3: "0 12 * * 1-5" would trigger an event at noon on Monday through Friday of each week.</p> <p>Example 4: "@weekly" would trigger an event at midnight once a week on Sunday.</p>

time	(Optional) Must be specified if a timer type other than CRON is specified. Must not be specified if the CRON timer type is specified. For watchdog and countdown timers, the number of seconds and milliseconds until the timer expires; for the absolute timer, the calendar time of the expiration time. Time is specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999. An absolute expiration date is the number of seconds and milliseconds since January 1, 1970. If the date specified has already passed, the timer expires immediately.
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Table 203: Time and Date When CRON Events Will Be Triggered

Field	Allowed Values
minute	0-59
hour	0-23
day of month	1-31
month	1-12 (or names, see below)
day of week	0-7 (0 or 7 is Sun, or names; see the table titled "Special Strings for cron_entry")

Table 204: Special Strings for cron_entry

String	Meaning
@yearly	Trigger once a year, "0 0 1 1 *".
@annually	Same as @yearly.
@monthly	Trigger once a month, "0 0 1 * *".
@weekly	Trigger once a week, "0 0 * * 0".
@daily	Trigger once a day, "0 0 * * *".
@midnight	Same as @daily.
@hourly	Trigger once an hour, "0 * * * *".

Result String

None

Set _cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"timer_type %s timer_time_sec %ld timer_time_msec %ld"
"timer_remain_sec %ld timer_remain_msec %ld"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
timer_type	Type of the timer. Can be one of the following: <ul style="list-style-type: none"> • watchdog • countdown • absolute
timer_time_sec timer_time_msec	Time when the timer expired.

Event Type	Description
timer_remain_sec timer_remain_msec	The remaining time before the next expiration.

See Also

event_register_timer_subscriber

event_register_timer_subscriber

Registers for a timer event as a subscriber. Use this Tcl command extension to identify the name of the timer to which the event timer, as a subscriber, wants to subscribe. The event timer depends on another policy or another process to actually manipulate the timer. For example, let policyB act as a timer subscriber policy, but policyA (although it does not need to be a timer policy) uses register_timer, timer_arm, or timer_cancel Tcl command extensions to manipulate the timer referenced in policyB.

Syntax

```
event_register_timer_subscriber watchdog|countdown|absolute|cron
name ? [queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

Arguments

watchdog	(Mandatory) Watchdog timer.
countdown	(Mandatory) Countdown timer.
absolute	(Mandatory) Absolute timer.
cron	(Mandatory) CRON timer.
name	(Mandatory) Name of the timer.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>



Note An EEM policy that registers for a timer event or a counter event can act as both publisher and subscriber.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"timer_type %s timer_time_sec %ld timer_time_msec %ld"
"timer_remain_sec %ld timer_remain_msec %ld"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.

Event Type	Description
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
timer_type	Type of the timer. Can be one of the following: <ul style="list-style-type: none"> • watchdog • countdown • absolute
timer_time_sec timer_time_msec	Time when the timer expired.
timer_remain_sec timer_remain_msec	The remaining time before the next expiration.

See Also

event_register_timer

event_register_track

Registers for a report event from the Cisco IOS Object Tracking subsystem. Use this Tcl command extension to trigger a policy on the basis of a Cisco IOS Object Tracking subsystem report for a specified object number.

Syntax

```
event_register_track ? [tag ?] [state up|down|any] [queue_priority low|normal|high|last]
[maxrun ?]
[nice 0|1]
```

Arguments

? (represents a number)	(Mandatory) Tracked object number in the range from 1 to 500, inclusive.
tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
state	(Optional) Specifies that the tracked object transition will cause an event to be raised. If up is specified, an event will be raised when the tracked object transitions from a down state to an up state. If down is specified, an event will be raised when the tracked object transitions from an up state to a down state. If any is specified, an event will be raised when the tracked object transitions to or from any state.

queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>
maxrun	<p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>
nice	<p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>

If an optional argument is not specified, the event matches all possible values of the argument.

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"track_number {%u} track_state {%s}"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event ID.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.

Event Type	Description
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
track_number	Number of the tracked object that caused the event to be triggered.
track_state	State of the tracked object when the event was triggered; valid states are up or down.

event_register_wdsysmon

Registers for a Watchdog system monitor event. Use this Tcl command extension to register for a composite event which is a combination of several subevents or conditions. For example, you can use this command to register for the combination of conditions wherein the CPU usage of a certain process is over 80 percent and the memory used by the process is greater than 50 percent of its initial allocation. This Tcl command extension is supported only in Software Modularity images.

Syntax

```
event_register_wdsysmon [tag ?] [timewin ?]
[sub12_op and|or|andnot]
[sub23_op and|or|andnot]
[sub34_op and|or|andnot]
[sub1 subevent-description]
[sub2 subevent-description]
[sub3 subevent-description]
[sub4 subevent-description] [node ?]
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

Each argument is position independent.



Note Operator definitions: and (logical and operation), or (logical or operation), andnot (logical and not operation). For example, "sub12_op and" is defined as raise an event when subevent 1 and subevent 2 are true; "sub23_op or" is defined as raise an event when the condition specified in sub12_op is true or subevent 3 is true. The logic can be diagrammed using: if (((sub1 sub12_op sub2) sub23_op sub3) sub34_op sub4) is TRUE, raise event

Arguments

tag	(Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.
timewin	(Optional) Time window within which all of the subevents have to occur in order for an event to be generated (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999).

sub12_op	(Optional) Combination operator for comparison between subevent 1 and subevent 2.
sub23_op	(Optional) Combination operator for comparison between subevent 1 and 2 and subevent 3.
sub34_op	(Optional) Combination operator for comparison between subevent 1 and 2 and subevent 3 and subevent 4.
sub1	(Optional) Indicates that subevent 1 is specified.
subevent-description	(Optional) Syntax for the subevent.
sub2	(Optional) Indicates that subevent 2 is specified.
sub3	(Optional) Indicates that subevent 3 is specified.
sub4	(Optional) Indicates that subevent 4 is specified.
node	<p>(Optional) The node name to be monitored for deadlock conditions is a string that consists of the word "node" followed by two fields separated by a slash character using the following format:</p> <pre>node<slot-number>/<cpu-number></pre> <p>The slot-number is the hardware slot number. The cpu-number is the hardware CPU number. For example, the SP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be specified as node0/0. The RP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be addressed as node0/1. If the node argument is not specified, the default node specification is the local node on which the registration is done.</p>
queue_priority	<p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> • queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels. • queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority. • queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels. • queue_priority last--Specifies that the script is to be queued at the lowest priority level. <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>Note The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p>

maxrun	(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.
nice	(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.

Subevents

The syntax of subevent descriptions can be one of seven cases.

For arguments in subevent description, the following constraints apply on the value of number arguments:

- For dispatch_mgr, val must be an integer between 0 and 4294967295, inclusive.
- For cpu_proc and cpu_tot, val must be an integer between 0 and 100, inclusive.
- For mem_proc, mem_tot_avail, and mem_tot_used, if is_percent is FALSE, val must be an integer between 0 and 4294967295, inclusive.

1. deadlock procname ?

Arguments

procname	(Mandatory) A regular expression that specifies the process name that you wish to monitor for deadlock conditions. This subevent will ignore the time window even if it is given.
----------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2. dispatch_mgr [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [period ?]

Arguments

procname	(Optional) A regular expression that specifies the process name that you wish to monitor for dispatch_manager status.
op	(Optional) Comparison operator used to compare the collected number of events with the specified value; if true, an event will be raised.
val	(Optional) The value with which the number of events that have occurred should be compared.
period	(Optional) The time period for the number of events that have occurred (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.

3. cpu_proc [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [period ?]

Arguments

procname	(Optional) A regular expression that specifies the process name that you wish to monitor for CPU utilization conditions.
op	(Optional) Comparison operator used to compare the collected CPU usage sample percentage with the specified percentage value; if true, an event will be raised.
val	(Optional) The percentage value with which the average CPU usage during the sample period should be compared.
period	(Optional) The time period for averaging the collection of samples (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.

4. cpu_tot [op gt|ge|eq|ne|lt|le] [val ?] [period ?]

Arguments

op	(Optional) Comparison operator used to compare the collected total system CPU usage sample percentage with the specified percentage value; if true, an event will be raised.
val	(Optional) The percentage value with which the average CPU usage during the sample period should be compared.
period	(Optional) The time period for averaging the collection of samples (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.

5. mem_proc [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period ?]

Arguments

procname	(Optional) A regular expression that specifies the process name that you wish to monitor for memory usage.
op	(Optional) Comparison operator used to compare the collected memory used with the specified value; if true, an event will be raised.
val	(Optional) A percentage or an absolute value specified in kilobytes. A percentage represents the difference between the oldest sample in the specified time period and the latest sample. If memory usage has increased from 150 KB to 300 KB within the time period, the percentage increase is 100. This is the value with which the measured value should be compared.
is_percent	(Optional) If TRUE, the percentage value is collected and compared. Otherwise, the absolute value is collected and compared.

period	(Optional) If is_percent is set to TRUE, the time period for the percentage to be computed. Otherwise, the time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.
--------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

6. mem_tot_avail [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period ?]

Arguments

op	(Optional) Comparison operator used to compare the collected available memory with the specified value; if true, an event will be raised.
val	(Optional) A percentage or an absolute value specified in kilobytes. A percentage represents the difference between the oldest sample in the specified time period and the latest sample. If available memory usage has decreased from 300 KB to 150 KB within the time period, the percentage decrease is 50. This is the value with which the measured value should be compared.
is_percent	(Optional) If TRUE, the percentage value is collected and compared. Otherwise, the absolute value is collected and compared.
period	(Optional) If is_percent is set to TRUE, the time period for the percentage to be computed. Otherwise, the time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.

7. mem_tot_used [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period ?]

Arguments

op	(Optional) Comparison operator used to compare the collected used memory with the specified value; if true, an event will be raised.
val	(Optional) A percentage or an absolute value specified in kilobytes. A percentage represents the difference between the oldest sample in the specified time period and the latest sample. If memory usage has increased from 150 KB to 300 KB within the time period, the percentage increase is 100. This is the value with which the measured value should be compared.
is_percent	(Optional) If TRUE, the percentage value is collected and compared. Otherwise, the absolute value is collected and compared.

period	<p>(Optional) If is_percent is set to TRUE, the time period for the percentage to be computed. Otherwise, the time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.</p> <p>Note This argument is mandatory if is_percent is set to TRUE; otherwise, it is optional.</p>
--------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Result String

None

Set_cerrno

No

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"num_subs %u"
```

Event Type	Description
event_id	Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.
event_type	Type of event.
event_type_string	An ASCII string that represents the name of the event for this event type.
event_pub_sec event_pub_msec	The time, in seconds and milliseconds, when the event was published to the EEM.
num_subs	Subevent number.

Where the subevent info string is for a deadlock subevent:

```
"{type %s num_entries %u entries {entry 1, entry 2, ...}}"
```

Subevent Type	Description
type	Type of wdsysmon subevent.
num_entries	Number of processes and threads in the deadlock.
entries	Information of processes and threads in the deadlock.

Where each entry is:

```
"{node {%s} procname {%s} pid %u tid %u state %s b_node %s b_procname %s b_pid %u
b_tid %u}"
```

Assume that the entry describes the scenario in which Process A thread m is blocked on process B thread n:

Subevent Type	Description
node	Name of the node that process A thread m is on.
procname	Name of process A.
pid	Process ID of process A.
tid	Thread ID of process A thread m.
state	Thread state of process A thread m. Can be one of the following: <ul style="list-style-type: none"> • STATE_CONDVAR • STATE_DEAD • STATE_INTR • STATE_JOIN • STATE_MUTEX • STATE_NANOSLEEP • STATE_READY • STATE_RECEIVE • STATE_REPLY • STATE_RUNNING • STATE_SEM • STATE_SEND • STATE_SIGSUSPEND • STATE_SIGWAITINFO • STATE_STACK • STATE_STOPPED • STATE_WAITPAGE • STATE_WAITTHREAD
b_node	Name of the node that process B thread is on.
b_procname	Name of process B.
b_pid	Process ID of process B.

Subevent Type	Description
b_tid	Thread ID of process B thread n; 0 means that process A thread m is blocked on all threads of process B.

For dispatch_mgr Subevent

```
"{type %s node {%s} procname {%s} pid %u value %u sec %ld msec %ld}"
```

Subevent Type	Description
type	Type of wdsysmon subevent.
node	Name of the node that the POSIX process is on.
procname	POSIX process name for this subevent.
pid	POSIX process ID for this subevent. Note The three fields above describe the owner process of this dispatch manager.
value	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the number of events processed by the dispatch manager is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the total number of events processed by this dispatch manager is in the given time window.
sec msec	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the sec and msec variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.

For cpu_proc Subevent

```
"{type %s node {%s} procname {%s} pid %u value %u sec %ld msec %ld}"
```

Subevent Type	Description
type	Type of wdsysmon subevent.
node	Name of the node that the POSIX process is on.
procname	POSIX process name for this subevent.
pid	POSIX process ID for this subevent. Note The three fields above describe the process whose CPU utilization is being monitored.

Subevent Type	Description
value	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the process CPU utilization is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged process CPU utilization is in the given time window.
sec msec	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the sec and msec variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.

For cpu_tot Subevent

```
"{type %s node {%s} value %u sec %ld msec %ld}"
```

Subevent Type	Description
type	Type of wdsysmon subevent.
node	Name of the node on which the total CPU utilization is being monitored.
value	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the total CPU utilization is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged total CPU utilization is in the given time window.
sec msec	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the sec and msec variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.

For mem_proc Subevent

```
"{type %s node {%s} procname {%s} pid %u is_percent %s value %u diff %d sec %ld msec %ld}"
```

Subevent Type	Description
type	Type of wdsysmon subevent.
node	Name of the node that the POSIX process is on.
procname	POSIX process name for this subevent.
pid	POSIX process ID for this subevent.
	Note The three fields above describe the process whose memory usage is being monitored.

Subevent Type	Description
is_percent	Can be either TRUE or FALSE. TRUE means that the value is a percentage value; FALSE means that the value is an absolute value (may be an averaged value).
value	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the process used memory is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged process used memory utilization is in the given time window.
Subevent Type	Description
diff	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the diff is the percentage difference between the first process used memory sample ever collected and the latest process used memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the diff is the percentage difference between the oldest and latest process used memory utilization in the specified time window.
sec msec	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the sec and msec variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.

If the **is_percent** argument is FALSE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **value** is the process used memory in the latest sample.
- **diff** is 0.
- **sec** and **msec** are both 0.

If the **is_percent** argument is FALSE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **value** is the averaged process used memory sample value in the specified time window.
- **diff** is 0.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest samples in this time window.

If the **is_percent** argument is TRUE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **value** is 0.
- **diff** is the percentage difference between the oldest and latest process used memory samples in the specified time window.
- **sec** and **msec** are the actual time difference between the time stamps of the oldest and latest process used memory samples in this time window.

If the **is_percent** argument is TRUE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **value** is 0.
- **diff** is the percentage difference between the first process used memory sample ever collected and the latest process used memory sample.
- **sec** and **msec** are the actual time difference between the time stamps of the first process used memory sample ever collected and the latest process used memory sample.

For mem_tot_avail Subevent

```
"{type %s node {%s} is_percent %s used %u avail %u diff %d sec %ld msec %ld}"
```

Subevent Type	Description
type	Type of wdsysmon subevent.
node	Name of the node for which the total available memory is being monitored.
is_percent	Can be either TRUE or FALSE. TRUE means that the value is a percentage value; FALSE means that the value is an absolute value (may be an averaged value).
used	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the total used memory is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged total used memory utilization is in the given time window.
avail	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the avail is in the latest total available memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the avail is the total available memory utilization in the specified time window.
diff	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the diff is the percentage difference between the first total available memory sample ever collected and the latest total available memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the diff is the percentage difference between the oldest and latest total available memory utilization in the specified time window.
sec msec	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, they are the actual time difference between the time stamps of the oldest and latest samples in this time window.

If the **is_percent** argument is FALSE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is the total used memory in the latest sample.
- **avail** is the total available memory in the latest sample.
- **diff** is 0.

- **sec** and **msec** are both 0.

If the **is_percent** argument is FALSE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is 0.
- **avail** is the averaged total available memory sample value in the specified time window.
- **diff** is 0.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total available memory samples in this time window.

If the **is_percent** argument is TRUE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.
- **diff** is the percentage difference between the oldest and latest total available memory samples in the specified time window.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total available memory samples in this time window.

If the **is_percent** argument is TRUE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.
- **diff** is the percentage difference between the first total available memory sample ever collected and the latest total available memory sample.
- **sec** and **msec** are the actual time difference between the time stamps of the first total available memory sample ever collected and the latest total available memory sample.

For mem_tot_used Subevent

```
"{type %s node %s} is_percent %s used %u avail %u diff %d sec %ld msec %ld}"
```

Subevent Type	Description
type	Type of wdsysmon subevent.
node	Name of the node for which the total used memory is being monitored.
is_percent	Can be either TRUE or FALSE. TRUE means that the value is a percentage value; FALSE means that the value is an absolute value (may be an averaged value).

Subevent Type	Description
used	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the total used memory is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged total used memory utilization is in the given time window.
avail	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the avail is in the latest total used memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the avail is the total used memory utilization in the specified time window.
diff	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, the diff is the percentage difference between the first total used memory sample ever collected and the latest total used memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the diff is the percentage difference between the oldest and latest total used memory utilization in the specified time window.
sec msec	If the sec and msec variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the sec and msec variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.

If the **is_percent** argument is FALSE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is the total used memory in the latest sample,
- **avail** is the total available memory in the latest sample,
- **diff** is 0,
- **sec** and **msec** are both 0,

If the **is_percent** argument is FALSE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is the averaged total used memory sample value in the specified time window,
- **avail** is 0,
- **diff** is 0,
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total used memory samples in this time window,

If the **is_percent** argument is TRUE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.

- **diff** is the percentage difference between the oldest and latest total used memory samples in the specified time window.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total used memory samples in this time window.

If the **is_percent** argument is TRUE, and the sec and msec arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.
- **diff** is the percentage difference between the first total used memory sample ever collected and the latest total used memory sample.
- **sec** and **msec** are the actual time difference between the time stamps of the first total used memory sample ever collected and the latest total used memory sample.



Note Inside a subevent description, each argument is position independent.



CHAPTER 94

EEM Event Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



Note For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.



Note Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

- [event_completion](#), on page 2017
- [event_completion_with_wait](#), on page 2018
- [event_publish](#), on page 2019
- [event_wait](#), on page 2022

event_completion

Sends a notification to the EEM server that the policy is done servicing the event that triggered it. The event only takes a single argument which is the **return_code** of this event instance.

Syntax

```
event_completion status ?
```

Arguments

status	(Mandatory) Exit status (return_code) of this event instance. A value of zero indicates no error and any other integer value indicates an error.
--------	--------------------------------------------------------------------------------------------------------------------------------------------------

Result String

None

Set_cerrno

No

event_completion_with_wait

The **event_completion_with_wait** command combines the two commands **event_completion** and **event_wait** into a single command for ease of use.

The **event_completion** command sends a notification to the EEM server that the policy is done servicing the event that triggered it. The event only takes a single argument which is the **return_code** of this event instance.

The **event_wait** places the Tcl policy into a sleep state. When the Tcl policy receives a new signal announcing a new event, the policy is placed into a wake state and again returns to a sleep state. This loop continues. If **event_wait** policy is invoked before **event_completed** policy, an error results and the policy exits.

Syntax

```
event_completion_with_wait status ? [refresh_vars]
```

Arguments

status	(Mandatory) exit_status (return_code) of this event instance. A value of zero indicates no error. Any other integer value indicates an error.
refresh_vars	(Optional) Indicates whether built-in and environment variables should be updated (refreshed) from the EEM Policy Director during this event instance.

Result String

None

Set_cerrno

Yes

Sample Usage

Here is a similar example as above using this single command:

```
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

```

set i 1
while {1 == 1} { # Start high performance policy loop
  array set arr_einfo [event_reqinfo]
  if {$_cerno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
      $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
  }
  action_syslog msg "event $i serviced" priority info
  if {$i == 5} {
    action_syslog msg "Exiting after servicing 5 events" priority info
    exit 0
  }
  incr i
  array set _event_state_arr [event_completion_with_wait status 0 refresh_vars 1]
  if {$_event_state_arr(event_state) != 0} {
    action_syslog msg "Exiting: failed event_state " \
      "$event_state_arr(event_state)" priority info
    exit 0
  }
}
}

```



Note The running configuration output is the same as the `event_publishTcl` command.

event_publish

Publishes an application-specific event.

Syntax

```
event_publish sub_system ? type ? [arg1 ?] [arg2 ?] [arg3 ?] [arg4 ?]
```

Arguments

sub_system	(Mandatory) Number assigned to the EEM policy that published the application-specific event. Number is set to 798 because all other numbers are reserved for Cisco use.
type	(Mandatory) Event subtype within the specified component. The sub_system and type arguments uniquely identify an application event. Must be an integer between 1 and 4294967295, inclusive.
[arg1 ?]-[arg4 ?]	(Optional) Four pieces of application event publisher string data.

Result String

None

Set_cerno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

Sample Usage

This example demonstrates how to use the **event_publish** Tcl command extension to execute a script *n* times repeatedly to perform some function (for example, to measure the amount of CPU time taken by a given group of Tcl statements). This example uses two Tcl scripts.

Script1 publishes a type 9999 EEM event to cause Script2 to run for the first time. Script1 is registered as a none event and is run using the Cisco IOS CLI **event manager run** command. Script2 is registered as an EEM application event of type 9999, and this script checks to see if the application publish arg1 data (the iteration number) exceeds the EEM environment variable test_iterations value. If the test_iterations value is exceeded, the script writes a message and exits; otherwise the script executes the remaining statements and reschedules another run. To measure the CPU utilization for Script2, use a value of test_iterations that is a multiple of 10 to calculate the amount of average CPU time used by Script2.

To run the Tcl scripts, enter the following Cisco IOS commands:

```
configure terminal
  event manager environment test_iterations 100
  event manager policy script1.tcl
  event manager policy script2.tcl
end
event manager run script1.tcl
```

The Tcl script Script2 will be executed 100 times. If you execute the script without the extra processing and derive the average CPU utilization, and then add the extra processing and repeat the test, you can subtract the former CPU utilization from the later CPU utilization to determine the average for the extra processing.

Script1 (script1.tcl)

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# Query the event info.
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
  set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
    $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
  error $result
}

action_syslog priority info msg "EEM application_publish test start"
if {$_cerrno != 0} {
  set result [format \
    "component=%s; subsys err=%s; posix err=%s;\n%s" \
    $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
  error $result
}

# Cause the first iteration to run.
event_publish sub_system 798 type 9999 arg1 0
if {$_cerrno != 0} {
  set result [format \
    "component=%s; subsys err=%s; posix err=%s;\n%s" \
    $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
```

```

    error $result
}

Script2 (script2.tcl)

::cisco::eem::event_register_appl sub_system 798 type 9999

# Check if all the required environment variables exist.
# If any required environment variable does not exist, print out an error msg and quit.
if (![info exists test_iterations]) {
    set result \
        "Policy cannot be run: variable test_iterations has not been set"
    error $result $errorInfo
}

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

# Query the event info.
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

# Data1 contains the arg1 value used to publish this event.
set iter $arr_einfo(data1)

# Use the arg1 info from the previous run to determine when to end.
if {$iter >= $test_iterations} {
    # Log a message.
    action_syslog priority info msg "EEM application_publish test end"
    if {$_cerrno != 0} {
        set result [format \
            "component=%s; subsys err=%s; posix err=%s;\n%s" \
            $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
        error $result
    }
    exit 0
}
set iter [expr $iter + 1]

# Log a message.
set msg [format "EEM application_publish test iteration %s" $iter]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

# Do whatever processing that you want to measure here.

# Cause the next iteration to run. Note that the iteration is passed to the
# next operation as arg1.
event_publish sub_system 798 type 9999 arg1 $iter
if {$_cerrno != 0} {
    set result [format \
        "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
}

```

event_wait

Places the Tcl policy into a sleep state. When the Tcl policy receives a new signal announcing a new event, the policy is placed into a wake state and again returns to a sleep state. This loop continues. If **event_wait** policy is invoked before **event_completed** policy, an error results and the policy exits.

Syntax

```
event_wait [refresh_vars]
```

Arguments

refresh_vars	(Optional) Indicates whether built-in and environment variables should be updated (refreshed) from the EEM Policy Director during this event instance.
--------------	--------------------------------------------------------------------------------------------------------------------------------------------------------

Result String

None

Set_cerrno

No

Sample Usage

The **event_wait** event detector returns an array type value with a single element named **event_state**. Event_state is a value sent back from the EEM Server indicating whether or not an error has occurred in processing the event. An example of an error here would be if the user configured **event_wait** before configuring **event_completion** when handling the event instance.

The following sample output shows the use of both **event_completion** and **event_wait**Tcl commands:

```
::cisco::eem::event_register_syslog tag e1 occurs 1 pattern CLEAR maxrun 0
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set i 1
while {1 == 1} { # Start high performance policy loop
  array set arr_einfo [event_reqinfo]
  if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
      $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
  }
  action_syslog msg "event $i serviced" priority info
  if {$i == 5} {
    action_syslog msg "Exiting after servicing 5 events" priority info
    exit 0
  }
  incr i
  event_completion status 0
  array set _event_state_arr [event_wait refresh_vars 0]
  if {$_event_state_arr(event_state) != 0} {
    action_syslog msg "Exiting: failed event_state " \
      " $_event_state_arr(event_state)" priority info
  }
}
```

```

        exit 0
    }
}

```

Here is an example of the running configuration:

```

Device#
01:00:44: %SYS-5-CONFIG_I: Configured from console by consoleclear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:00:49: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:49: %HA_EM-6-LOG: high_perf_example.tcl: event 1 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:00:53: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:53: %HA_EM-6-LOG: high_perf_example.tcl: event 2 serviced
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:00:56: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:56: %HA_EM-6-LOG: high_perf_example.tcl: event 3 serviced
Device#
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:00:59: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
01:00:59: %HA_EM-6-LOG: high_perf_example.tcl: event 4 serviced
01:00:59: %HA_EM-6-LOG: high_perf_example.tcl: Exiting after servicing 5 events
Device#
Device#
Device#copy tftp disk1:
Address or name of remote host [dirt]?
Source filename [user/eem_scripts/high_perf_example.tcl]?
Destination filename [high_perf_example.tcl]?
%Warning:There is a file already existing with this name
Do you want to over write? [confirm]
Accessing tftp://dirt/user/eem_scripts/high_perf_example.tcl...
Loading user/eem_scripts/high_perf_example.tcl from 192.0.2.19 (via FastEthernet0/0): !
[OK - 909 bytes]
909 bytes copied in 0.360 secs (2525 bytes/sec)
Device#
Device#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)#no event manager policy high_perf_example.tcl
Device(config)#event manager po high_perf_example.tcl
Device(config)#end
Device#
Device#
Device#
Device#
01:02:19: %SYS-5-CONFIG_I: Configured from console by consoleclear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:02:23: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
Device#
01:02:23: %HA_EM-6-LOG: high_perf_example.tcl: event 1 serviced
Device#

```

```

Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:26: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:26: %HA_EM-6-LOG: high_perf_example.tcl: event 2 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:29: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:29: %HA_EM-6-LOG: high_perf_example.tcl: event 3 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:33: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
01:02:33: %HA_EM-6-LOG: high_perf_example.tcl: event 4 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
Device#
01:02:36: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:36: %HA_EM-6-LOG: high_perf_example.tcl: event 5 serviced
01:02:36: %HA_EM-6-LOG: high_perf_example.tcl: Exiting after servicing 5 events
Device#

```

Also while an event has been serviced and is waiting for the next event to come in **show event manager policy active** command will display the following output:

```

Device#show event manager policy active
Key: p - Priority          :L - Low, H - High, N - Normal, Z - Last
      s - Scheduling node :A - Active, S - Standby
default class - 1 script event
no.  job id      p s status  time of event          event type          name
1    11         N A wait    Mon Oct20 14:15:24 2008  syslog
high_perf_example.tcl

```

In the above example the status is wait. This indicates that the policy is waiting for the next event to come in.



CHAPTER 95

EEM Library Debug Command Extensions

- [cli_debug](#), on page 2025
- [smtp_debug](#), on page 2025

cli_debug

Prints a command-line interface (CLI) debug statement to syslog. This Tcl command extension is used to print a CLI debug statement to syslog if the **debug event manager tcl cli_library** Cisco IOS CLI command is in effect.

Syntax

```
cli_debug spec_string debug_string
```

Arguments

spec_string	(Mandatory) The spec_string argument is used to indicate the type of debug statement.
debug_string	(Mandatory) The debug_string argument is used to indicate the debugging text.

Result String

None

Set_cerrno

No

smtp_debug

Prints a Simple Mail Transfer Protocol (SMTP) debug statement to syslog. This Tcl command extension prints a SMTP debug statement to syslog if the **debug event manager tcl smtp_library** Cisco IOS command-line interface (CLI) command is in effect.

Syntax

```
smtp_debug spec_string debug_string
```

Arguments

spec_string	(Mandatory) The spec_string argument is used to indicate the type of debug statement.
debug_string	(Mandatory) The debug_string argument is used to indicate the debugging text.

Result String

None

Set_cerrno

No



CHAPTER 96

EEM Multiple Event Support Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



Note For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.



Note Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

- [attribute](#), on page 2027
- [correlate](#), on page 2028
- [trigger](#), on page 2029

attribute

Specifies a complex event.

Syntax

```
attribute tag ? [occurs ?]
```

Arguments

tag	Specifies a tag using the <i>event-tag</i> argument that can be used with the attribute command to associate an event.
occurs	(Optional) Specifies the number of occurrences before an EEM event is triggered. If not specified, an EEM event is triggered on the first occurrence. The range is from 1 to 4294967295.

Result String

None

Set_cerrno

No

correlate

Builds a single complex event and allows boolean logic to relate events and tracked objects.

Syntax

```
correlate event ? track ? [andnot | and | or] event ? track ?
```

Arguments

event	Specifies the event that can be used with the trigger command to support multiple event statements within an script. If the event associated with the <i>event-tag</i> argument occurs for the number of times specified by the trigger command, the result is true. If not, the result is false.
track	Specifies the event object number for tracking. The range is from 1 to 500. If the tracked object is set, the result of the evaluation is true. If the tracked object is not set or is undefined, the result of the evaluation is false. This result is regardless of the state of the object.
<i>andnot</i>	(Optional) Specifies that if event 1 occurs the action is executed, and if event 2 and event 3 occur together the action is not executed.
and	(Optional) Specifies that if event 1 occurs the action is executed, and if event 2 and event 3 occur together the action is executed. Note When "and" is used to group events such as traps or syslog messages, then the default trigger occurrence window is three minutes.
or	(Optional) Specifies that if event 1 occurs the action is executed, or else if event 2 and event 3 occur together the action is executed.

Result String

None

Set _cerno

No

trigger

Specifies the multiple event configuration ability of Embedded Event Manager (EEM) events. A multiple event is one that can involve one or more event occurrences, one or more tracked object states, and a time period for the event to occur. The events are raised based on the specified parameters.

Syntax

```
trigger [occurs ?] [period ?] [period-start ?] [delay ?]
```

Arguments

occurs	(Optional) Specifies the number of times the total correlation occurs before an EEM event is raised. When a number is not specified, an EEM event is raised on the first occurrence. The range is from 1 to 4294967295.
period	(Optional) Time interval in seconds and optional milliseconds, during which the one or more occurrences must take place. This is specified in the format ssssssss[.mmm], where ssssssss must be an integer number representing seconds between 0 and 4294967295, inclusive and mmm represents milliseconds and must be an integer number between 0 to 999.
period-start	(Optional) Specifies the start of an event correlation window. If not specified, event monitoring is enabled after the first CRON period occurs.
delay	(Optional) Specifies the number of seconds and optional milliseconds after which an event will be raised if all the conditions are true (specified in the format ssssssss[.mmm], where ssssssss must be an integer number representing seconds between 0 and 4294967295, inclusive and mmm represents milliseconds and must be an integer number between 0 to 999).

Result String

None

Set _cerno

No

trigger



CHAPTER 97

EEM SMTP Library Command Extensions

All Simple Mail Transfer Protocol (SMTP) library command extensions belong to the `::cisco::lib` namespace.

To use this library, the user needs to provide an e-mail template file. The template file can include Tcl global variables so that the e-mail service and the e-mail text can be configured through the **event manager environment Cisco IOS** command-line interface (CLI) configuration command. There are commands in this library to substitute the global variables in the e-mail template file and to send the desired e-mail context with the To address, CC address, From address, and Subject line properly configured using the configured e-mail server.

E-Mail Template

The e-mail template file has the following format:



Note Based on RFC 2554, the SMTP e-mail server name--Mailservername-- can be in any one of the following template formats: `username:password@host`, `username@host`, or `host`.

```
Mailservername:<space><the list of candidate SMTP server addresses>
From:<space><the e-mail address of sender>
To:<space><the list of e-mail addresses of recipients>
Cc:<space><the list of e-mail addresses that the e-mail will be copied to>
Sourceaddr:<space><the IP addresses of the recipients>
Subject:<subject line>
<a blank line>
<body>
```



Note Note that the template normally includes Tcl global variables for configuration.

In a Tcl policy, the port number can be specified by a "Port" line in the e-mail template. If port is not specified, the default port of 25 is used.

Below is a sample e-mail template file:

```
Mailservername: $_email_server
From: $_email_from
To: $_email_to
Cc: $_email_cc
```

```
Sourceaddr: $_email_ipaddr
Port: <port number>
Subject: From router $routername: Process terminated
process name: $process_name
subsystem: $sub_system
exit status: $exit_status
respawn count: $respawn_count
```

- [smtp_send_email, on page 2032](#)
- [smtp_subst, on page 2033](#)

smtp_send_email

Given the text of an e-mail template file with all global variables already substituted, sends the e-mail out using Simple Mail Transfer Protocol (SMTP). The e-mail template specifies the candidate mail server addresses, To addresses, CC addresses, From address, subject line, and e-mail body.



Note A list of candidate e-mail servers can be provided so that the library will try to connect the servers on the list one by one until it can successfully connect to one of them.

Syntax

```
smtp_send_email text
```

Arguments

<code>text</code>	(Mandatory) The text of an e-mail template file with all global variables already substituted.
-------------------	------------------------------------------------------------------------------------------------

Result String

None

Set _cerrno

- Wrong 1st line format--Mailservername:list of server names.
- Wrong 2nd line format--From:from-address.
- Wrong 3rd line format--To:list of to-addresses.
- Wrong 4th line format--CC:list of cc-addresses.
- Error connecting to mail server:--\$sock closed by remote server (where \$sock is the name of the socket opened to the mail server).
- Error connecting to mail server:--\$sock reply code is \$k instead of the service ready greeting (where \$sock is the name of the socket opened to the mail server; \$k is the reply code of \$sock).
- Error connecting to mail server:--cannot connect to all the candidate mail servers.
- Error disconnecting from mail server:--\$sock closed by remote server (where \$sock is the name of the socket opened to the mail server).

Sample Scripts

After all needed global variables in the e-mail template are defined:

```
if [catch {smtp_subst [file join $tcl_library email_template_sm]} result] {
    puts stderr $result
    exit 1
}
if [catch {smtp_send_email $result} result] {
    puts stderr $result
    exit 1
}
```

smtp_subst

Given an e-mail template file e-mail_template, substitutes each global variable in the file by its user-defined value. Returns the text of the file after substitution.

Syntax

```
smtp_subst e-mail_template
```

Arguments

e-mail_template	(Mandatory) Name of an e-mail template file in which global variables need to be substituted by a user-defined value. An example filename could be /disk0://example.template which represents a file named example.template in a top-level directory on an ATA flash disk in slot 0.
-----------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Result String

The text of the e-mail template file with all the global variables substituted.

Set _cerrno

- cannot open e-mail template file
- cannot close e-mail template file

smtp_subst



CHAPTER 98

EEM System Information Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



Note All EEM system information commands--`sys_reqinfo_XXX`--have the Set _cerno section set to yes.



Note For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.



Note Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

- [sys_reqinfo_cli_freq](#), on page 2036
- [sys_reqinfo_cli_history](#), on page 2037
- [sys_reqinfo_cpu_all](#), on page 2037
- [sys_reqinfo_crash_history](#), on page 2038
- [sys_reqinfo_mem_all](#), on page 2039
- [sys_reqinfo_proc](#), on page 2040
- [sys_reqinfo_proc_all](#), on page 2042
- [sys_reqinfo_routename](#), on page 2042
- [sys_reqinfo_snmp](#), on page 2043
- [sys_reqinfo_syslog_freq](#), on page 2044

- [sys_reqinfo_syslog_history](#), on page 2045

sys_reqinfo_cli_freq

Queries the frequency information of all command-line interface (CLI) events.

Syntax

```
sys_reqinfo_cli_freq
```

Arguments

None

Result String

```
rec_list {{CLI frequency string 0},{CLI frequency str 1}, ...}
```

Where each CLI frequency string is:

```
time_sec %ld time_msec %ld match_count %u raise_count %u occurs %u period_sec %ld period_msec %ld
pattern {%s}
```

rec_list	Marks the start of the CLI event frequency list.
time_sec time_msec	Last time when this CLI event was raised.
match count	Number of times that a CLI command matches the pattern specified by this CLI event specification.
raise_count	Number of times that this CLI event was raised. The following fields are information about the CLI event specification: <ul style="list-style-type: none"> • sync--A "yes" means that event publish should be performed synchronously. The event detector will be notified when the Event Manager Server has completed publishing the event. The Event Manager Server will return a code that indicates whether or not the CLI command should be executed. • skip--A "yes" means that the CLI command should not be executed if the sync flag is not set.
occurs	Number of occurrences before an event is raised; if this argument is not specified, an event is raised on the first occurrence.
period_sec period_msec	Number of occurrences must occur within this number of POSIX timer units in order to raise event; if this argument is not specified, it does not apply.
pattern	Regular expression used to perform CLI command pattern matching.

Set_cerrno

Yes

sys_reqinfo_cli_history

Queries the history of command-line interface (CLI) commands.

Syntax

```
sys_reqinfo_cli_history
```

Arguments

None

Result String

```
rec_list {{CLI history string 0}, {CLI history str 1},...}
```

Where each CLI history string is:

```
time_sec %ld time_msec %ld cmd {%s}
```

rec_list	Marks the start of the CLI command history list.
time_sec time_msec	Time when the CLI command was run.
cmd	Text of the CLI command.

Set_cerrno

Yes

sys_reqinfo_cpu_all

Queries the CPU utilization of the top processes (both POSIX processes and IOS processes) during a specified time period and in a specified order. This Tcl command extension is supported only in Software Modularity images.

Syntax

```
sys_reqinfo_cpu_all order cpu_used [sec ?] [msec ?] [num ?]
```

Arguments

order	(Mandatory) Order used for sorting the CPU utilization of processes.
cpu_used	(Mandatory) Specifies that the average CPU utilization, for the specified time window, will be sorted in descending order.

sec msec	(Optional) The time period, in seconds and milliseconds, during which the average CPU utilization is calculated. Must be integers in the range from 0 to 4294967295. If not specified, or if both sec and msec are specified as 0, the most recent CPU sample is used.
num	(Optional) Number of entries from the top of the sorted list of processes to be displayed. Must be an integer in the range from 1 to 4294967295. Default value is 5.

Result String

```
rec_list {{process CPU info string 0},{process CPU info string 1}, ...}
```

Where each process CPU info string is:

```
pid %u name {%s} cpu_used %u
```

rec_list	Marks the start of the process CPU information list.
pid	Process ID.
name	Process name.
cpu_used	Specifies that if sec and msec are specified with a number greater than zero, the average percentage is calculated from the process CPU utilization during the specified time period. If sec and msec are both zero or not specified, the average percentage is calculated from the process CPU utilization in the latest sample.

Set_cerrno

Yes

sys_reqinfo_crash_history

Queries the crash information of all processes that have ever crashed. This Tcl command extension is supported only in Software Modularity images.

Syntax

```
sys_reqinfo_crash_history
```

Arguments

None

Result String

```
rec_list {{crash info string 0},{crash info string 1}, ...}
```

Where each crash info string is:

```
job_id %u name {%s} respawn_count %u fail_count %u dump_count %u
inst_id %d exit_status 0x%x exit_type %d proc_state {%s} component_id 0x%x
crash_time_sec %ld crash_time_msec %ld
```

job_id	System manager assigned job ID for the process. An integer between 1 and 4294967295, inclusive.
name	Process name.
respawn_count	Total number of restarts for the process.
fail_count	Number of restart attempts of the process. This count is reset to zero when the process is successfully restarted.
dump_count	Number of core dumps performed.
inst_id	Process instance ID.
exit_status	Last exit status of the process.
exit_type	Last exit type.
proc_state	Sysmgr process states. One of the following: error, forced_stop, hold, init, ready_to_run, run, run_rnode, stop, waitEOltimer, wait_rnode, wait_spawntimer, wait_tpl.
component_id	Version manager assigned component ID for the component to which the process belongs.
crash_time_sec crash_time_msec	Seconds and milliseconds since January 1, 1970, which represent the last time the process crashed.

Set _cerrno

Yes

sys_reqinfo_mem_all

Queries the memory usage of the top processes (both POSIX and IOS) during a specified time period and in a specified order. This Tcl command extension is supported only in Software Modularity images.

Syntax

```
sys_reqinfo_mem_all order allocates|increase|used [sec ?] [msec ?] [num ?]
```

Arguments

order	(Mandatory) Order used for sorting the memory usage of processes.
allocates	(Mandatory) Specifies that the memory usage is sorted by the number of process allocations during the specified time window, and in descending order.
increase	(Mandatory) Specifies that the memory usage is sorted by the percentage of process memory increase during the specified time window, and in descending order.
used	(Mandatory) Specifies that the memory usage is sorted by the current memory used by the process.

sec msec	(Optional) The time period, in seconds and milliseconds, during which the process memory usage is calculated. Must be integers in the range from 0 to 4294967295. If both sec and msec are specified and are nonzero, the number of allocations is the difference between the number of allocations in the oldest and latest samples collected in the time period. The percentage is calculated as the the percentage difference between the memory used in the oldest and latest samples collected in the time period. If not specified, or if both sec and msec are specified as 0, the first sample ever collected is used as the oldest sample; that is, the time period is set to be the time from startup until the current moment.
num	(Optional) Number of entries from the top of the sorted list of processes to be displayed. Must be an integer in the range from 1 to 4294967295. Default value is 5.

Result String

```
rec_list {{process mem info string 0},{process mem info string 1}, ...}
```

Where each process mem info string is:

```
pid %u name {%s} delta_allocs %d initial_alloc %u current_alloc %u percent_increase %d
```

rec_list	Marks the start of the process memory usage information list.
pid	Process ID.
name	Process name.
delta_allocs	Specifies the difference between the number of allocations in the oldest and latest samples collected in the time period.
initial_alloc	Specifies the amount of memory, in kilobytes, used by the process at the start of the time period.
current_alloc	Specifies the amount of memory, in kilobytes, currently used by the process.
percent_increase	Specifies the percentage difference between the memory used in the oldest and latest samples collected in the time period. The percentage difference can be expressed as current_alloc minus initial_alloc times 100 and divided by initial_alloc.

Set_cerrno

Yes

sys_reqinfo_proc

Queries the information about a single POSIX process. This Tcl command extension is supported only in Software Modularity images.

Syntax

```
sys_reqinfo_proc job_id ?
```


Arguments

job_id	(Mandatory) System manager assigned job ID for the process. Must be an integer between 1 and 4294967295, inclusive.
--------	---------------------------------------------------------------------------------------------------------------------

Result String

```
job_id %u component_id 0x%x name {%s} helper_name {%s} helper_path {%s} path {%s}
node_name {%s} is_respawn %u is_mandatory %u is_hold %u dump_option %d
max_dump_count %u respawn_count %u fail_count %u dump_count %u
last_respawn_sec %ld last_respawn_msec %ld inst_id %u proc_state %s
level %d exit_status 0x%x exit_type %d
```

job_id	System manager assigned job ID for the process. An integer between 1 and 4294967295, inclusive.
component_id	Version manager assigned component ID for the component to which the process belongs.
name	Process name.
helper_name	Helper process name.
helper_path	Executable path of the helper process.
path	Executable path of the process.
node_name	System manager assigned node name for the node to which the process belongs.
is_respawn	Flag that specifies that the process can be respawned.
is_mandatory	Flag that specifies that the process must be alive.
is_hold	Flag that specifies that the process is spawned until called by the API.
dump_option	Core dumping options.
max_dump_count	Maximum number of core dumping permitted.
respawn_count	Total number of restarts for the process.
fail_count	Number of restart attempts of the process. This count is reset to zero when the process is successfully restarted.
dump_count	Number of core dumps performed.
last_respawn_sec last_respawn_msec	Seconds and milliseconds in POSIX timer units since January 1, 1970, which represent the last time the process was started.
inst_id	Process instance ID.
proc_state	Sysmgr process states. One of the following: error, forced_stop, hold, init, ready_to_run, run, run_rnode, stop, waitEOltimer, wait_rnode, wait_spawntimer, wait_tpl.

level	Process run level.
exit_status	Last exit status of the process.
exit_type	Last exit type.

Set_cerrno

Yes

sys_reqinfo_proc_all

Queries the information of all POSIX processes. This Tcl command extension is supported only in Software Modularity images.

Syntax

```
sys_reqinfo_proc_all
```

Arguments

None

Result String

```
rec_list {{process info string 0}, {process info string 1},...}
```

Where each process info string is the same as the result string of the **sysreq_info_proc** Tcl command extension.

Set_cerrno

Yes

sys_reqinfo_routename

Queries the device name.

Syntax

```
sys_reqinfo_routename
```

Arguments

None

Result String

```
routename %s
```

Where routename is the name of the device.

Set _cerrno

Yes

sys_reqinfo_snmp

Queries the value of the entity specified by a Simple Network Management Protocol (SNMP) object ID.

Syntax

```
sys_reqinfo_snmp oid ? get_type exact|next
```

Arguments

oid	(Mandatory) SNMP OID in dot notation (for example, 1.3.6.1.2.1.2.1.0).
get_type	(Mandatory) Type of SNMP get operation that needs to be applied to the specified oid. If the get_type is "exact," the value of the specified oid is retrieved; if the get_type is "next," the value of the lexicographical successor to the specified oid is retrieved.

Result String

```
oid {%s} value {%s}
```

oid	SNMP OID.
value	Value string of the associated SNMP data element.

Set _cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 22)   FH_ENULLPTR   (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 37)   FH_ENOSNMPDATA (can't retrieve data from SNMP)
```

This error means that there was no data for the SNMP object type.

```
(_cerr_sub_err = 51)   FH_ESTATSTYP (invalid statistics data type)
```

This error means that the SNMP statistics data type was invalid.

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

sys_reqinfo_syslog_freq

Queries the frequency information of all syslog events.

Syntax

```
sys_reqinfo_syslog_freq
```

Arguments

None

Result String

```
rec_list {{event frequency string 0}, {log freq str 1}, ...}
```

Where each event frequency string is:

```
time_sec %ld time_msec %ld match_count %u raise_count %u occurs %u
period_sec %ld period_msec %ld pattern {%s}
```

time_sec time_msec	Seconds and milliseconds in POSIX timer units since January 1, 1970, which represent the time the last event was raised.
match_count	Number of times that a syslog message matches the pattern specified by this syslog event specification since event registration.
raise_count	Number of times that this syslog event was raised.
occurs	Number of occurrences needed in order to raise the event; if not specified, the event is raised on the first occurrence.
period_sec period_msec	Number of occurrences must occur within this number of POSIX timer units in order to raise the event; if not specified, the period check does not apply.
pattern	Regular expression used to perform syslog message pattern matching.

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 9)    FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 22)    FH_ENULLPTR    (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 45)    FH_ESEQNUM    (sequence or workset number out of sync)
```

This error means that the event detector sequence or workset number was invalid.

```
(_cerr_sub_err = 46)    FH_EREGEMPTY    (registration list is empty)
```

This error means that the event detector registration list was empty.

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL    (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

sys_reqinfo_syslog_history

Queries the history of the specified syslog message.

Syntax

```
sys_reqinfo_syslog_history
```

Arguments

None

Result String

```
rec_list {{log hist string 0}, {log hist str 1}, ...}
```

Where each log hist string is:

```
time_sec %ld time_msec %ld msg {%s}
```

time_sec time_msec	Seconds and milliseconds since January 1, 1970, which represent the time the message was logged.
msg	Syslog message.

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 44)    FH_EHISTEMPTY  (history list is empty)
```

This error means that the history list was empty.

```
(_cerr_sub_err = 45)    FH_ESEQNUM  (sequence or workset number out of sync)
```

This error means that the event detector sequence or workset number was invalid.

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

This error means that the event detector was unavailable.



CHAPTER 99

EEM Utility Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



Note For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.



Note Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

- [appl_read](#), on page 2048
- [appl_reqinfo](#), on page 2048
- [appl_setinfo](#), on page 2049
- [counter_modify](#), on page 2050
- [description](#), on page 2051
- [fts_get_stamp](#), on page 2052
- [register_counter](#), on page 2053
- [register_timer](#), on page 2054
- [timer_arm](#), on page 2056
- [timer_cancel](#), on page 2057
- [unregister_counter](#), on page 2058

appl_read

Reads Embedded Event Manager (EEM) application volatile data. This Tcl command extension provides support for reading EEM application volatile data. EEM application volatile data can be published by a Cisco software process that uses the EEM application publish API. EEM application volatile data cannot be published by an EEM policy.



Note Currently there are no Cisco software processes that publish application volatile data.

Syntax

```
appl_read name ? length ?
```

Arguments

name	(Mandatory) Name of the application published string data.
length	(Mandatory) Length of the string data to read. Must be an integer number between 1 and 4294967295, inclusive.

Result String

```
data %s
```

Where data is the application published string data to be read.

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 7)    FH_ENOSUCHKEY  (could not find key)
```

This error means that the application event detector info key or other ID was not found.

```
(_cerr_sub_err = 9)    FH_EMEMORY  (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

appl_reqinfo

Retrieves previously saved information from the Embedded Event Manager (EEM). This Tcl command extension provides support for retrieving information from EEM that has been previously saved with a unique

key, which must be specified in order to retrieve the information. Note that retrieving the information deletes it from EEM. It must be resaved if it is to be retrieved again.

Syntax

```
appl_reqinfo key ?
```

Arguments

key	(Mandatory) The string key of the data.
-----	-----------------------------------------

Result String

```
data %s
```

Where data is the application string data to be retrieved.

Set _cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 7)    FH_ENOSUCHKEY  (could not find key)
```

This error means that the application event detector info key or other ID was not found.

appl_setinfo

Saves information in the Embedded Event Manager (EEM). This Tcl command extension provides support for saving information in the Embedded Event Manager that can be retrieved later by the same policy or by another policy. A unique key must be specified. This key allows the information to be retrieved later.

Syntax

```
appl_setinfo key ? data ?
```

Arguments

key	(Mandatory) The string key of the data.
data	(Mandatory) The application string data to save.

Result String

None

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 8)    FH_EDUPLICATEKEY    (duplicate appl info key)
```

This error means that the application event detector info key or other ID was a duplicate.

```
(_cerr_sub_err = 9)    FH_EMEMORY    (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 34)   FH_EMAXLEN    (maximum length exceeded)
```

This error means that the object length or number exceeded the maximum.

```
(_cerr_sub_err = 43)   FH_EBADLENGTH    (bad API length)
```

This error means that the API message length was invalid.

counter_modify

Modifies a counter value.

Syntax

```
counter_modify event_id ? val ? op nop|set|inc|dec
```

Arguments

event_id	(Mandatory) The counter event ID returned by the register_counter Tcl command extension. Must be an integer between 0 and 4294967295, inclusive.
val	(Mandatory) Note Mandatory except when the op nop argument value combination is specified. <ul style="list-style-type: none"> • If op is set, this argument represents the counter value that is to be set. • If op is inc, this argument is the value by which to increment the counter. • If op is dec, this argument is the value by which to decrement the counter.

op	<p>(Mandatory)</p> <ul style="list-style-type: none"> • nop--Retrieves the current counter value. • set--Sets the counter value to the given value. • inc--Increments the counter value by the given value. • dec--Decrements the counter value by the given value.
----	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Result String

```
val_remain %d
```

Where val_remain is the current value of the counter.

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 11)   FH_ENOSUCHESID  (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22)   FH_ENULLPTR  (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 30)   FH_ECTBADOPER  (bad counter threshold operator)
```

This error means that the counter event detector set or modify operator was invalid.

description

Provides a brief description of the registered policy.

Syntax

```
description ?
```

Arguments

line	(Optional) Brief description of the policy consisting of 1 to 240 characters.
------	-------------------------------------------------------------------------------

Result String

None

Set_cerrno

Yes

Sample Usage

The description statement is entered by the author of the policy. It can appear before or after any event registration statement in Tcl. The policy can have only one description.



Note Registration of a policy with more than one description statement will fail.

The following example shows how a brief description is provided for the **event_register_syslog** policy:

```
::cisco::eem::description "This Tcl command looks for the word count in syslog messages."
::cisco::eem::event_register_syslog tag 1 ...
::cisco::eem::event_register_snmp_object tag 2 ...
::cisco::eem::trigger {
    ::cisco::eem::correlate event 1 and event 2
    ::cisco::eem::attribute tag 1 occurs 1
    ::cisco::eem::attribute tag 2 occurs 1
}
```

fts_get_stamp

Returns the time period elapsed since the last software boot. Use this Tcl command extension to return the number of nanoseconds since boot in an array "nsec nnnn" where nnnn is the number of nanoseconds.

Syntax

```
fts_get_stamp
```

Arguments

None

Result String

```
nsec %d
```

Where nsec is the number of nanoseconds since boot.

Set_cerrno

No

register_counter

Registers a counter and returns a counter event ID. This Tcl command extension is used by a counter publisher to perform this registration before using the event ID to manipulate the counter.

Syntax

```
register_counter name ?
```

Arguments

name	(Mandatory) The name of the counter to be manipulated.
-------------	--------------------------------------------------------

Result String

```
event_id %d
event_spec_id %d
```

Where `event_id` is the counter event ID for the specified counter; it can be used to manipulate the counter by the **unregister_counter** or **counter_modify** Tcl command extensions. The `event_spec_id` argument is the event specification ID for the specified counter.

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX `errno` value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 4)    FH_EINITONCE  (Init() is not yet done, or done twice.)
```

This error means that the request to register the specific event was made before the EEM event detector had completed its initialization.

```
(_cerr_sub_err = 6)    FH_EBADEVENTTYPE (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 9)    FH_EMEMORY    (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 10)   FH_ECORRUPT  (internal EEM API context is corrupt)
```

This error means that the internal EEM API context structure is corrupt.

```
(_cerr_sub_err = 11)   FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 12)    FH_ENOSUCHEID (unknown event ID)
```

This error means that the event ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 16)    FH_EBADFMPPTR (bad ptr to fh_p data structure)
```

This error means that the context pointer that is used with each EEM API call is incorrect.

```
(_cerr_sub_err = 17)    FH_EBADADDRESS (bad API control block address)
```

This error means that a control block address that was passed in the EEM API was incorrect.

```
(_cerr_sub_err = 22)    FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 25)    FH_ESUBSEXCEED (number of subscribers exceeded)
```

This error means that the number of timer or counter subscribers exceeded the maximum.

```
(_cerr_sub_err = 26)    FH_ESUBSIDXINV (invalid subscriber index)
```

This error means that the subscriber index was invalid.

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56)    FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

register_timer

Registers a timer and returns a timer event ID. This Tcl command extension is used by a timer publisher to perform this registration before using the event ID to manipulate the timer if it does not use the **event_register_timer** command extension to register as a publisher and subscriber.

Syntax

```
register_timer watchdog|countdown|absolute|cron name ?
```

Arguments

name	(Mandatory) The name of the timer to be manipulated.
------	------------------------------------------------------

Result String

```
event_id %u
```

Where `event_id` is the timer event ID for the specified timer (can be used to manipulate the timer by the `timer_arm` or `timer_cancel` command extensions).

Set `_cerrno`

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX `errno` value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 4)    FH_EINITONCE  (Init() is not yet done, or done twice.)
```

This error means that the request to register the specific event was made before the EEM event detector had completed its initialization.

```
(_cerr_sub_err = 6)    FH_EBADEVENTTYPE (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 9)    FH_EMEMORY   (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 10)   FH_ECORRUPT  (internal EEM API context is corrupt)
```

This error means that the internal EEM API context structure is corrupt.

```
(_cerr_sub_err = 11)   FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 16)   FH_EBADFMPPTR (bad ptr to fh_p data structure)
```

This error means that the context pointer that is used with each EEM API call is incorrect.

```
(_cerr_sub_err = 17)   FH_EBADADDRESS (bad API control block address)
```

This error means that a control block address that was passed in the EEM API was incorrect.

```
(_cerr_sub_err = 22)   FH_ENULLPTR   (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 25)   FH_ESUBSEXCEED (number of subscribers exceeded)
```

This error means that the number of timer or counter subscribers exceeded the maximum.

```
(_cerr_sub_err = 26)   FH_ESUBSIDXINV (invalid subscriber index)
```

This error means that the subscriber index was invalid.

```
(_cerr_sub_err = 54)   FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56)    FH_EFDCONNERR    (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

timer_arm

Arms a timer. The type could be CRON, watchdog, countdown, or absolute.

Syntax

```
timer_arm event_id ? cron_entry ?|time ?
```

Arguments

event_id	(Mandatory) The timer event ID returned by the register_timer command extension. Must be an integer between 0 and 4294967295, inclusive.
cron_entry	(Mandatory) Must exist if the timer type is CRON. Must not exist for other types of timer. CRON timer specification uses the format of the CRON table entry.
time	(Mandatory) Must exist if the timer type is not CRON. Must not exist if the timer type is CRON. For watchdog and countdown timers, the number of seconds and milliseconds until the timer expires; for an absolute timer, the calendar time of the expiration time (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). An absolute expiration date is the number of seconds and milliseconds since January 1, 1970. If the date specified has already passed, the timer expires immediately.

Result String

```
sec_remain %ld msec_remain %ld
```

Where sec_remain and msec_remain are the remaining time before the next expiration of the timer.



Note A value of 0 will be returned for the sec_remain and msec_remain arguments if the timer type is CRON.

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.


```
(_cerr_sub_err = 6)    FH_EBADEVENTTYPE (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 9)    FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 11)   FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 12)   FH_ENOSUCHEID (unknown event ID)
```

This error means that the event ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22)   FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 27)   FH_ETMDELAYZR (zero delay time)
```

This error means that the time specified to arm a timer was zero.

```
(_cerr_sub_err = 42)   FH_ENOTREGISTERED (request for event spec that is unregistered)
```

This error means that the event was not registered.

```
(_cerr_sub_err = 54)   FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56)   FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

timer_cancel

Cancels a timer.

Syntax

```
timer_cancel event_id ?
```

Arguments

event_id	(Mandatory) The timer event ID returned by the register_timer command extension. Must be an integer between 0 and 4294967295, inclusive.
----------	-------------------------------------------------------------------------------------------------------------------------------------------------

Result String

```
sec_remain %ld msec_remain %ld
```

Where sec_remain and msec_remain are the remaining time before the next expiration of the timer.



Note A value of 0 will be returned for sec_remain and msec_remain if the timer type is CRON .

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 6)    FH_EBADEVENTTYPE    (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 7)    FH_ENOSUCHKEY    (could not find key)
```

This error means that the application event detector info key or other ID was not found.

```
(_cerr_sub_err = 11)   FH_ENOSUCHESID    (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 12)   FH_ENOSUCHEID    (unknown event ID)
```

This error means that the event ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22)   FH_ENULLPTR    (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 54)   FH_EFDUNAVAIL    (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56)   FH_EFDCONNERR    (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

unregister_counter

Unregisters a counter. This Tcl command extension is used by a counter publisher to unregister a counter that was previously registered with the **register_counter** Tcl command extension.

Syntax

```
unregister_counter event_id ? event_spec_id ?
```

Arguments

event_id	(Mandatory) Counter event ID returned by the register_counter command extension. Must be an integer between 0 and 4294967295, inclusive.
event_spec_id	(Mandatory) Counter event specification ID for the specified counter returned by the register_counter command extension. Must be an integer between 0 and 4294967295, inclusive.

Result String

None

Set_cerrno

Yes

```
(_cerr_sub_err = 2)    FH_ESYSERR    (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 9)    FH_EMEMORY    (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 11)   FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22)   FH_ENULLPTR    (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 26)   FH_ESUBSIDXINV (invalid subscriber index)
```

This error means that the subscriber index was invalid.

```
(_cerr_sub_err = 54)   FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56)   FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

■ unregister_counter



PART **XII**

VLAN

- [Configuring VTP, on page 2063](#)
- [Configuring VLANs, on page 2085](#)
- [Configuring VLAN Trunks, on page 2099](#)
- [Configuring VMPS, on page 2117](#)
- [Configuring Voice VLANs, on page 2129](#)
- [Configuring Private VLANs, on page 2137](#)



CHAPTER 100

Configuring VTP

- [Finding Feature Information, on page 2063](#)
- [Prerequisites for VTP, on page 2063](#)
- [Restrictions for VTP, on page 2064](#)
- [Information About VTP, on page 2064](#)
- [How to Configure VTP, on page 2071](#)
- [Monitoring VTP, on page 2081](#)
- [Configuration Examples for VTP, on page 2081](#)
- [Where to Go Next, on page 2082](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for VTP

Before you create VLANs, you must decide whether to use the VLAN Trunking Protocol (VTP) in your network. Using VTP, you can make configuration changes centrally on one or more devices and have those changes automatically communicated to all the other devices in the network. Without VTP, you cannot send information about VLANs to other devices.

VTP is designed to work in an environment where updates are made on a single device and are sent through VTP to other devices in the domain. It does not work well in a situation where multiple updates to the VLAN database occur simultaneously on devices in the same domain, which would result in an inconsistency in the VLAN database.

The device supports a total of 1000 VLANs. However, the number of routed ports, SVIs, and other configured features affects the usage of the device hardware. If the device is notified by VTP of a new VLAN and the device is already using the maximum available hardware resources, it sends a message that there are not

enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the device and that this trunk port is connected to the trunk port of another device. Otherwise, the device cannot receive any VTP advertisements.

Restrictions for VTP



Note Before adding a VTP client device to a VTP domain, always verify that its VTP configuration revision number is lower than the configuration revision number of the other devices in the VTP domain. Devices in a VTP domain always use the VLAN configuration of the device with the highest VTP configuration revision number. If you add a device that has a revision number higher than the revision number in the VTP domain, it can erase all VLAN information from the VTP server and VTP domain.

The following are restrictions for configuring VTPs:

- 1K VLAN is supported only on switches running the LAN Base image with the lanbase-default template set.
- To avoid warning messages of high CPU utilization with a normal-range VLAN configuration, we recommended to have no more than 256 VLANs.

In such cases, approximately 10 access interfaces or 5 trunk interfaces can flap simultaneously with negligible impact to CPU utilization (if there are more interfaces that flap simultaneously, then CPU usage may be excessively high.)

Information About VTP

VTP

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can cause several problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). VTP version 3 supports the entire VLAN range (VLANs 1 to 4094). Extended range VLANs (VLANs 1006 to 4094) are supported only in VTP version 3.

You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured in the domain.

VTP Domain

A VTP domain (also called a VLAN management domain) consists of one device or several interconnected devices under the same administrative responsibility sharing the same VTP domain name. A device can be in only one VTP domain. You make global VLAN configuration changes for the domain.

By default, the device is in the VTP no-management-domain state until it receives an advertisement for a domain over a trunk link (a link that carries the traffic of multiple VLANs) or until you configure a domain name. Until the management domain name is specified or learned, you cannot create or modify VLANs on a VTP server, and VLAN information is not propagated over the network.

If the device receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The device then ignores advertisements with a different domain name or an earlier configuration revision number.

When you make a change to the VLAN configuration on a VTP server, the change is propagated to all devices in the VTP domain. VTP advertisements are sent over all IEEE trunk connections, including IEEE 802.1Q. VTP dynamically maps VLANs with unique names and internal index associates across multiple LAN types. Mapping eliminates excessive device administration required from network administrators.

If you configure a device for VTP transparent mode, you can create and modify VLANs, but the changes are not sent to other devices in the domain, and they affect only the individual device. However, configuration changes made when the device is in this mode are saved in the device running configuration and can be saved to the device startup configuration file.

VTP Modes

Table 205: VTP Modes

VTP Mode	Description
VTP server	<p>In VTP server mode, you can create, modify, and delete VLANs, and specify other configuration (such as the VTP version) for the entire VTP domain. VTP servers advertise their VLAN configurations to other devices in the same VTP domain and synchronize their VLAN configurations with other devices' configurations received over trunk links.</p> <p>VTP server is the default mode.</p> <p>In VTP server mode, VLAN configurations are saved in NVRAM. If the device detects a failure to save a configuration to NVRAM, VTP mode automatically changes from server mode to client mode. If the device cannot be returned to VTP server mode until the NVRAM is functioning.</p>
VTP client	<p>A VTP client functions like a VTP server and transmits and receives VTP updates on its trunks, but cannot create, change, or delete VLANs on a VTP client. VLANs are configured on another device in the same VTP domain that is in server mode.</p> <p>In VTP versions 1 and 2 in VTP client mode, VLAN configurations are not saved in NVRAM. In VTP version 3, VLAN configurations are saved in NVRAM in client mode.</p>

VTP Mode	Description
VTP transparent	<p>VTP transparent devices do not participate in VTP. A VTP transparent device does not advertise its VTP configuration and does not synchronize its VLAN configuration based on received advertisements. However, in VTP version 2 or version 3, transparent devices do forward VTP advertisements that they receive from other devices through their trunk interfaces. You can create, modify, and delete VLANs on a device in VTP transparent mode.</p> <p>In VTP versions 1 and 2, the device must be in VTP transparent mode when you create private VLANs. When private VLANs are configured, you should not change the VTP mode from transparent to client or server mode. VTP version 3 also supports private VLANs in client and server modes. When private VLANs are configured, do not change the VTP mode from transparent to client or server mode.</p> <p>When the device is in VTP transparent mode, the VTP and VLAN configurations are saved in NVRAM. However, they are not advertised to other devices. In this mode, VTP mode and domain name are saved in the running configuration, and you can save this information in the device startup configuration file by using the copy running-config startup-config privileged EXEC command.</p>
VTP off	A device in VTP off mode functions in the same manner as a VTP transparent device, except that it does not forward VTP advertisements on trunks.

VTP Advertisements

Each device in the VTP domain sends periodic global configuration advertisements from each trunk port to a reserved multicast address. Neighboring devices receive these advertisements and update their VTP and VLAN configurations as necessary.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the switch stack and that this trunk port is connected to the trunk port of another switch. Otherwise, the switch cannot receive any VTP advertisements.

VTP advertisements distribute this global domain information:

- VTP domain name
- VTP configuration revision number
- Update identity and update timestamp
- MD5 digest VLAN configuration, including maximum transmission unit (MTU) size for each VLAN
- Frame format

VTP advertisements distribute this VLAN information for each configured VLAN:

- VLAN IDs (including IEEE 802.1Q)
- VLAN name
- VLAN type
- VLAN state
- Additional VLAN configuration information specific to the VLAN type

In VTP version 3, VTP advertisements also include the primary server ID, an instance number, and a start index.

VTP Version 2

If you use VTP in your network, you must decide which version of VTP to use. By default, VTP operates in version 1.

VTP version 2 supports these features that are not supported in version 1:

- Token Ring support—VTP version 2 supports Token Ring Bridge Relay Function (TrBRF) and Token Ring Concentrator Relay Function (TrCRF) VLANs.
- Unrecognized Type-Length-Value (TLV) support—A VTP server or client propagates configuration changes to its other trunks, even for TLVs it is not able to parse. The unrecognized TLV is saved in NVRAM when the device is operating in VTP server mode.
- Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent device inspects VTP messages for the domain name and version and forwards a message only if the version and domain name match. Although VTP version 2 supports only one domain, a VTP version 2 transparent device forwards a message only when the domain name matches.
- Consistency Checks—In VTP version 2, VLAN consistency checks (such as VLAN names and values) are performed only when you enter new information through the CLI or SNMP. Consistency checks are not performed when new information is obtained from a VTP message or when information is read from NVRAM. If the MD5 digest on a received VTP message is correct, its information is accepted.

VTP Version 3

VTP version 3 supports these features that are not supported in version 1 or version 2:

- Enhanced authentication—You can configure the authentication as **hidden** or **secret**. When **hidden**, the secret key from the password string is saved in the VLAN database file, but it does not appear in plain text in the configuration. Instead, the key associated with the password is saved in hexadecimal format in the running configuration. You must reenter the password if you enter a takeover command in the domain. When you enter the **secret** keyword, you can directly configure the password secret key.
- Support for extended range VLAN (VLANs 1006 to 4094) database propagation—VTP versions 1 and 2 propagate only VLANs 1 to 1005.



Note VTP pruning still applies only to VLANs 1 to 1005, and VLANs 1002 to 1005 are still reserved and cannot be modified.

- Private VLAN support.
- Support for any database in a domain—In addition to propagating VTP information, version 3 can propagate Multiple Spanning Tree (MST) protocol database information. A separate instance of the VTP protocol runs for each application that uses VTP.

- VTP primary server and VTP secondary servers—A VTP primary server updates the database information and sends updates that are honored by all devices in the system. A VTP secondary server can only back up the updated VTP configurations received from the primary server to its NVRAM.

By default, all devices come up as secondary servers. You can enter the **vtp primary** privileged EXEC command to specify a primary server. Primary server status is only needed for database updates when the administrator issues a takeover message in the domain. You can have a working VTP domain without any primary servers. Primary server status is lost if the device reloads or domain parameters change, even when a password is configured on the device.

VTP Pruning

VTP pruning increases network available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to reach the destination devices. Without VTP pruning, a device floods broadcast, multicast, and unknown unicast traffic across all trunk links within a VTP domain even though receiving devices might discard them. VTP pruning is disabled by default.

VTP pruning blocks unneeded flooded traffic to VLANs on trunk ports that are included in the pruning-eligible list. Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning eligible device trunk ports. If the VLANs are configured as pruning-ineligible, the flooding continues. VTP pruning is supported in all VTP versions.

With VTP versions 1 and 2, when you enable pruning on the VTP server, it is enabled for the entire VTP domain. In VTP version 3, you must manually enable pruning on each device in the domain. Making VLANs pruning-eligible or pruning-ineligible affects pruning eligibility for those VLANs on that trunk only (not on all devices in the VTP domain).

VTP pruning takes effect several seconds after you enable it. VTP pruning does not prune traffic from VLANs that are pruning-ineligible. VLAN 1 and VLANs 1002 to 1005 are always pruning-ineligible; traffic from these VLANs cannot be pruned. Extended-range VLANs (VLAN IDs higher than 1005) are also pruning-ineligible.

VTP Configuration Guidelines

VTP Configuration Requirements

When you configure VTP, you must configure a trunk port so that the device can send and receive VTP advertisements to and from other devices in the domain.

VTP versions 1 and 2 do not support private VLANs. VTP version 3 does support private VLANs. If you configure private VLANs, the device must be in VTP transparent mode. When private VLANs are configured on the device, do not change the VTP mode from transparent to client or server mode.

VTP Settings

The VTP information is saved in the VTP VLAN database. When VTP mode is transparent, the VTP domain name and mode are also saved in the device running configuration file, and you can save it in the device startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. You must use this command if you want to save VTP mode as transparent, even if the device resets.

When you save VTP information in the device startup configuration file and reboot the device, the device configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration do not match the VLAN database, the domain name and VTP mode and configuration for VLAN IDs 1 to 1005 use the VLAN database information.

Domain Names for Configuring VTP

When configuring VTP for the first time, you must always assign a domain name. You must configure all devices in the VTP domain with the same domain name. Devices in VTP transparent mode do not exchange VTP messages with other devices, and you do not need to configure a VTP domain name for them.



Note If the NVRAM and DRAM storage is sufficient, all devices in a VTP domain should be in VTP server mode.



Caution Do not configure a VTP domain if all devices are operating in VTP client mode. If you configure the domain, it is impossible to make changes to the VLAN configuration of that domain. Make sure that you configure at least one device in the VTP domain for VTP server mode.

Passwords for the VTP Domain

You can configure a password for the VTP domain, but it is not required. If you do configure a domain password, all domain devices must share the same password and you must configure the password on each device in the management domain. Devices without a password or with the wrong password reject VTP advertisements.

If you configure a VTP password for a domain, a device that is booted without a VTP configuration does not accept VTP advertisements until you configure it with the correct password. After the configuration, the device accepts the next VTP advertisement that uses the same password and domain name in the advertisement.

If you are adding a new device to an existing network with VTP capability, the new device learns the domain name only after the applicable password has been configured on it.



Caution When you configure a VTP domain password, the management domain does not function properly if you do not assign a management domain password to each device in the domain.

VTP Version

Follow these guidelines when deciding which VTP version to implement:

- All devices in a VTP domain must have the same domain name, but they do not need to run the same VTP version.
- A VTP version 2-capable device can operate in the same VTP domain as a device running VTP version 1 if version 2 is disabled on the version 2-capable device (version 2 is disabled by default).

- If a device running VTP version 1, but capable of running VTP version 2, receives VTP version 3 advertisements, it automatically moves to VTP version 2.
- If a device running VTP version 3 is connected to a device running VTP version 1, the VTP version 1 device moves to VTP version 2, and the VTP version 3 device sends scaled-down versions of the VTP packets so that the VTP version 2 device can update its database.
- A device running VTP version 3 cannot move to version 1 or 2 if it has extended VLANs.
- Do not enable VTP version 2 on a device unless all of the devices in the same VTP domain are version-2-capable. When you enable version 2 on a device, all of the version-2-capable devices in the domain enable version 2. If there is a version 1-only device, it does not exchange VTP information with devices that have version 2 enabled.
- Cisco recommends placing VTP version 1 and 2 devices at the edge of the network because they do not forward VTP version 3 advertisements.
- If there are TrBRF and TrCRF Token Ring networks in your environment, you must enable VTP version 2 or version 3 for Token Ring VLAN switching to function properly. To run Token Ring and Token Ring-Net, disable VTP version 2.
- VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4094). You must configure these VLANs manually on each device. VTP version 3 supports extended-range VLANs and support for extended range VLAN database propagation.
- When a VTP version 3 device trunk port receives messages from a VTP version 2 device, it sends a scaled-down version of the VLAN database on that particular trunk in VTP version 2 format. A VTP version 3 device does not send VTP version 2-formatted packets on a trunk unless it first receives VTP version 2 packets on that trunk port.
- When a VTP version 3 device detects a VTP version 2 device on a trunk port, it continues to send VTP version 3 packets, in addition to VTP version 2 packets, to allow both kinds of neighbors to coexist on the same trunk.
- A VTP version 3 device does not accept configuration information from a VTP version 2 or version 1 device.
- Two VTP version 3 regions can only communicate in transparent mode over a VTP version 1 or version 2 region.
- Devices that are only VTP version 1 capable cannot interoperate with VTP version 3 devices.
- VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4094). You must manually configure these VLANs on each device.

Default VTP Configuration

The following table shows the default VTP configuration.

Table 206: Default VTP Configuration

Feature	Default Setting
VTP domain name	Null
VTP mode (VTP version 1 and version 2)	Server

Feature	Default Setting
VTP mode (VTP version 3)	The mode is the same as the mode in VTP version 1 or 2 before conversion to version 3.
VTP version	Version 1
MST database mode	Transparent
VTP version 3 server type	Secondary
VTP password	None
VTP pruning	Disabled

How to Configure VTP

Configuring VTP Mode

You can configure VTP mode as one of these:

- VTP server mode—In VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.
- VTP client mode—In VTP client mode, you cannot change its VLAN configuration. The client device receives VTP updates from a VTP server in the VTP domain and then modifies its configuration accordingly.
- VTP transparent mode—In VTP transparent mode, VTP is disabled on the device. The device does not send VTP updates and does not act on VTP updates received from other device. However, a VTP transparent device running VTP version 2 does forward received VTP advertisements on its trunk links.
- VTP off mode—VTP off mode is the same as VTP transparent mode except that VTP advertisements are not forwarded.

When you configure a domain name, it cannot be removed; you can only reassign a device to a different domain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **vtp mode** {client | server | transparent | off} {vlan | mst | unknown}
5. **vtp password** *password*
6. **end**
7. **show vtp status**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vtp domain <i>domain-name</i> Example: Switch(config)# vtp domain eng_group	Configures the VTP administrative-domain name. The name can be 1 to 32 characters. All devices operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name. This command is optional for modes other than server mode. VTP server mode requires a domain name. If the device has a trunk connection to a VTP domain, the device learns the domain name from the VTP server in the domain. You should configure the VTP domain before configuring other VTP parameters.
Step 4	vtp mode {client server transparent off} {vlan mst unknown} Example: Switch(config)# vtp mode server	Configures the device for VTP mode (client, server, transparent, or off). <ul style="list-style-type: none"> • vlan—The VLAN database is the default if none are configured. • mst—The multiple spanning tree (MST) database. • unknown—An unknown database type.
Step 5	vtp password <i>password</i> Example: Switch(config)# vtp password mypassword	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. If you configure a VTP password, the VTP domain does not function properly if you do not assign the same password to each device in the domain.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show vtp status Example: <pre>Switch# show vtp status</pre>	Verifies your entries in the <i>VTP Operating Mode</i> and the <i>VTP Domain Name</i> fields of the display.
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves the configuration in the startup configuration file. Only VTP mode and domain name are saved in the device running configuration and can be copied to the startup configuration file.

Configuring a VTP Version 3 Password

You can configure a VTP version 3 password on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp version 3**
4. **vtp password** *password* [**hidden** | **secret**]
5. **end**
6. **show vtp password**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	vtp version 3 Example: <pre>Switch(config)# vtp version 3</pre>	Enables VTP version 3 on the device. The default is VTP version 1.

	Command or Action	Purpose
Step 4	vtp password <i>password</i> [hidden secret] Example: <pre>Switch(config)# vtp password mypassword hidden</pre>	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. <ul style="list-style-type: none"> • (Optional) hidden—Saves the secret key generated from the password string in the nvram:vlan.dat file. If you configure a takeover by configuring a VTP primary server, you are prompted to reenter the password. • (Optional) secret—Directly configures the password. The secret password must contain 32 hexadecimal characters.
Step 5	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show vtp password Example: <pre>Switch# show vtp password</pre>	Verifies your entries. The output appears like this: VTP password: 89914640C8D90868B6A0D8103847A733
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a VTP Version 3 Primary Server

When you configure a VTP server as a VTP primary server, the takeover operation starts.

SUMMARY STEPS

1. **vtp version 3**
2. **vtp primary** [**vlan** | **mst**] [**force**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	vtp version 3 Example: <pre>Switch(config)# vtp version 3</pre>	Enables VTP version 3 on the device. The default is VTP version 1.

	Command or Action	Purpose
Step 2	vtp primary [vlan mst] [force] Example: <pre>Switch# vtp primary vlan force</pre>	<p>Changes the operational state of a device from a secondary server (the default) to a primary server and advertises the configuration to the domain. If the device password is configured as hidden, you are prompted to reenter the password.</p> <ul style="list-style-type: none"> • (Optional) vlan—Selects the VLAN database as the takeover feature. This is the default. • (Optional) mst—Selects the multiple spanning tree (MST) database as the takeover feature. • (Optional) force—Overwrites the configuration of any conflicting servers. If you do not enter force, you are prompted for confirmation before the takeover.

Enabling the VTP Version

VTP version 2 and version 3 are disabled by default.

- When you enable VTP version 2 on a device, every VTP version 2-capable device in the VTP domain enables version 2. To enable VTP version 3, you must manually configure it on each device.
- With VTP versions 1 and 2, you can configure the version only on devices in VTP server or transparent mode. If a device is running VTP version 3, you can change to version 2 when the device is in client mode if no extended VLANs exist, no private VLANs exist, and no hidden password was configured.



Caution VTP version 1 and VTP version 2 are not interoperable on devices in the same VTP domain. Do not enable VTP version 2 unless every device in the VTP domain supports version 2.

- In TrCRF and TrBRF Token Ring environments, you must enable VTP version 2 or VTP version 3 for Token Ring VLAN switching to function properly. For Token Ring and Token Ring-Net media, disable VTP version 2.



Caution In VTP version 3, both the primary and secondary servers can exist on an instance in the domain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp version {1 | 2 | 3}**
4. **end**
5. **show vtp status**

6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vtp version {1 2 3} Example: Switch(config)# vtp version 2	Enables the VTP version on the device. The default is VTP version 1.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show vtp status Example: Switch# show vtp status	Verifies that the configured VTP version is enabled.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling VTP Pruning

Before you begin

VTP pruning is not designed to function in VTP transparent mode. If one or more devices in the network are in VTP transparent mode, you should do one of these actions:

- Turn off VTP pruning in the entire network.

- Turn off VTP pruning by making all VLANs on the trunk of the device upstream to the VTP transparent device pruning ineligible.

To configure VTP pruning on an interface, use the **switchport trunk pruning vlan** interface configuration command. VTP pruning operates when an interface is trunking. You can set VLAN pruning-eligibility, whether or not VTP pruning is enabled for the VTP domain, whether or not any given VLAN exists, and whether or not the interface is currently trunking.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp pruning**
4. **end**
5. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	vtp pruning Example: <pre>Switch(config)# vtp pruning</pre>	Enables pruning in the VTP administrative domain. By default, pruning is disabled. You need to enable pruning on only one device in VTP server mode.
Step 4	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show vtp status Example: <pre>Switch# show vtp status</pre>	Verifies your entries in the <i>VTP Pruning Mode</i> field of the display.

Configuring VTP on a Per-Port Basis

With VTP version 3, you can enable or disable VTP on a per-port basis. You can enable VTP only on ports that are in trunk mode. Incoming and outgoing VTP traffic are blocked, not forwarded.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **vtp**
5. **end**
6. **show running-config interface** *interface-id*
7. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet0/1	Identifies an interface, and enters interface configuration mode.
Step 4	vtp Example: Switch(config-if)# vtp	Enables VTP on the specified port.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show running-config interface <i>interface-id</i> Example: <pre>Switch# show running-config interface gigabitethernet 1/0/1</pre>	Verifies the change to the port.
Step 7	show vtp status Example: <pre>Switch# show vtp status</pre>	Verifies the configuration.

Adding a VTP Client to a VTP Domain

Follow these steps to verify and reset the VTP configuration revision number on a device *before* adding it to a VTP domain.

Before you begin

Before adding a VTP client to a VTP domain, always verify that its VTP configuration revision number is *lower* than the configuration revision number of the other devices in the VTP domain. Devices in a VTP domain always use the VLAN configuration of the device with the highest VTP configuration revision number. With VTP versions 1 and 2, adding a device that has a revision number higher than the revision number in the VTP domain can erase all VLAN information from the VTP server and VTP domain. With VTP version 3, the VLAN information is not erased.

You can use the **vtp mode transparent** global configuration command to disable VTP on the device and then to change its VLAN information without affecting the other devices in the VTP domain.

SUMMARY STEPS

1. **enable**
2. **show vtp status**
3. **configure terminal**
4. **vtp domain *domain-name***
5. **end**
6. **show vtp status**
7. **configure terminal**
8. **vtp domain *domain-name***
9. **end**
10. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show vtp status Example: Switch# show vtp status	Checks the VTP configuration revision number. If the number is 0, add the device to the VTP domain. If the number is greater than 0, follow these substeps: <ul style="list-style-type: none"> • Write down the domain name. • Write down the configuration revision number. • Continue with the next steps to reset the device configuration revision number.
Step 3	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 4	vtp domain domain-name Example: Switch(config)# vtp domain domain123	Changes the domain name from the original one displayed in Step 1 to a new name.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode. The VLAN information on the device is updated and the configuration revision number is reset to 0.
Step 6	show vtp status Example: Switch# show vtp status	Verifies that the configuration revision number has been reset to 0.
Step 7	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 8	vtp domain domain-name Example:	Enters the original domain name on the device

	Command or Action	Purpose
	Switch(config)# vtp domain domain012	
Step 9	end Example: Switch(config)# end	Returns to privileged EXEC mode. The VLAN information on the device is updated.
Step 10	show vtp status Example: Switch# show vtp status	(Optional) Verifies that the domain name is the same as in Step 1 and that the configuration revision number is 0.

Monitoring VTP

This section describes commands used to display and monitor the VTP configuration.

You monitor VTP by displaying VTP configuration information: the domain name, the current VTP revision, and the number of VLANs. You can also display statistics about the advertisements sent and received by the device.

Table 207: VTP Monitoring Commands

Command	Purpose
show vtp counters	Displays counters about VTP messages.
show vtp devices [conflict]	Displays information about all VTP version 3 devices with conflicting primary domain names. The conflict keyword does not display information when the device is not a primary server.
show vtp interface [interface-id]	Displays VTP status and configuration for the specified interface.
show vtp password	Displays the VTP password. The form of the password is displayed only if the hidden keyword was entered and the password is not the default.
show vtp status	Displays the VTP device configuration.

Configuration Examples for VTP

Example: Configuring a Switch as the Primary Server

This example shows how to configure a device as the primary server for the VLAN database (the default) when a hidden or secret password was configured:

```
Switch# vtp primary vlan
Enter VTP password: mypassword
This switch is becoming Primary server for vlan feature in the VTP domain

VTP Database Conf Switch ID      Primary Server Revision System Name
-----
VLANDB          Yes  00d0.00b8.1400=00d0.00b8.1400 1          stp7

Do you want to continue (y/n) [n]? y
```

Example: Configuring Switch as VTP Server

This example shows how to configure the switch as a VTP server with the domain name *eng_group* and the password *mypassword*:

```
Switch(config)# vtp domain eng_group
Setting VTP domain name to eng_group.

Switch(config)# vtp mode server
Setting device to VTP Server mode for VLANS.

Switch(config)# vtp password mypassword
Setting device VLAN database password to mypassword.
Switch(config)# end
```

Example: Enabling VTP on the Interface

To enable VTP on the interface, use the **vtp** interface configuration command. To disable VTP on the interface, use the **no vtp** interface configuration command.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# vtp
Switch(config-if)# end
```

Example: Creating the VTP Password

The follow is an example of creating the VTP password.

```
Switch(config)# vtp password mypassword hidden
Generating the secret associated to the password.
Switch(config)# end
Switch# show vtp password
VTP password: 89914640C8D90868B6A0D8103847A733
```

Where to Go Next

After configuring VTP, you can configure the following:

- VLANs
- VLAN Trunking
- VLAN Membership Policy Server (VMPS)
- Voice VLANs



CHAPTER 101

Configuring VLANs

- [Finding Feature Information, on page 2085](#)
- [Prerequisites for VLANs, on page 2085](#)
- [Restrictions for VLANs, on page 2085](#)
- [Information About VLANs, on page 2086](#)
- [How to Configure VLANs, on page 2090](#)
- [Monitoring VLANs, on page 2097](#)
- [Configuration Examples, on page 2098](#)
- [Where to Go Next, on page 2098](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for VLANs

The following are prerequisites and considerations for configuring VLANs:

- Before you create VLANs, you must decide whether to use VLAN Trunking Protocol (VTP) to maintain global VLAN configuration for your network.
- The switch supports 1000 VLANs in VTP client, server, and transparent modes.

Restrictions for VLANs

The following are restrictions for configuring VLANs:

- To avoid warning messages of high CPU utilization with a normal-range VLAN configuration, we recommend that you have no more than 256 VLANs. In such cases, approximately 10 access interfaces

or 5 trunk interfaces can flap simultaneously with negligible impact to CPU utilization (if there are more interfaces that flap simultaneously, then CPU usage may be excessively high.)

Information About VLANs

Logical Networks

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment. Any device port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router or a device supporting fallback bridging. Because a VLAN is considered a separate logical network, it contains its own bridge Management Information Base (MIB) information and can support its own implementation of spanning tree.

VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the device is assigned manually on an interface-by-interface basis. When you assign device interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed.

The device can route traffic between VLANs by using device virtual interfaces (SVIs). An SVI must be explicitly configured and assigned an IP address to route traffic between VLANs.

Supported VLANs

The switch supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs.

VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). In these versions, the switch must be in VTP transparent mode when you create VLAN IDs from 1006 to 4094. Cisco IOS Release 12.2(52)SE and later support VTP version 3. VTP version 3 supports the entire VLAN range (VLANs 1 to 4094). Extended range VLANs (VLANs 1006 to 4094) are supported only in VTP version 3. You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured in the domain.

The switch supports per-VLAN spanning-tree plus (PVST+) or rapid PVST+ with a maximum of 128 spanning-tree instances. One spanning-tree instance is allowed per VLAN. The switch supports only IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that specifies the kind of traffic the port carries and the number of VLANs to which it can belong.

When a port belongs to a VLAN, the device learns and manages the addresses associated with the port on a per-VLAN basis.

Table 208: Port Membership Modes and Characteristics

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN.	VTP is not required. If you do not want VTP to globally propagate information, set the VTP mode to transparent. To participate in VTP, there must be at least one trunk port on the device connected to a trunk port of a second device.
Trunk (IEEE 802.1Q) : <ul style="list-style-type: none"> IEEE 802.1Q— Industry-standard trunking encapsulation. 	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. You can also modify the pruning-eligible list to block flooded traffic to VLANs on trunk ports that are included in the list.	VTP is recommended but not required. VTP maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP exchanges VLAN configuration messages with other devices over trunk links.
Dynamic access	A dynamic-access port can belong to one VLAN (VLAN ID 1 to 4094) and is dynamically assigned by a VLAN Member Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch, for example, but never a Catalyst switch. The Catalyst switch is a VMPS client. You can have dynamic-access ports and trunk ports on the same device, but you must connect the dynamic-access port to an end station or hub and not to another device.	VTP is required. Configure the VMPS and the client with the same VTP domain name. To participate in VTP, at least one trunk port on the device must be connected to a trunk port of a second device .
Voice VLAN	A voice VLAN port is an access port attached to a Cisco IP Phone, configured to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.	VTP is not required; it has no effect on a voice VLAN.

VLAN Configuration Files

Configurations for VLAN IDs 1 to 1005 are written to the `vlan.dat` file (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The `vlan.dat` file is stored in flash memory. If the VTP mode is transparent, they are also saved in the device running configuration file.

You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the device, the device configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration, and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the domain name and VTP mode and configuration for the VLAN IDs 1 to 1005 use the VLAN database information.
- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for VLAN IDs 1 to 1005 use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4094.
- From image 15.0(02)SE6, on vtp transparent and off modes, vlans get created from startup-config even if they are not applied to the interface.



Note Ensure that you delete the vlan.dat file along with the configuration files before you reset the switch configuration using **write erase** command. This ensures that the switch reboots correctly on a reset.

Normal-Range VLAN Configuration Guidelines

Normal-range VLANs are VLANs with IDs from 1 to 1005.

VTP 1 and 2 only support normal-range VLANs.

Follow these guidelines when creating and modifying normal-range VLANs in your network:

- Normal-range VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- VLAN configurations for VLANs 1 to 1005 are always saved in the VLAN database. If the VTP mode is transparent, VTP and VLAN configurations are also saved in the device running configuration file.
- If the device is in VTP server or VTP transparent mode, you can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)
- With VTP versions 1 and 2, the device supports VLAN IDs 1006 through 4094 only in VTP transparent mode (VTP disabled). These are extended-range VLANs and configuration options are limited. Extended-range VLANs created in VTP transparent mode are not saved in the VLAN database and are not propagated. VTP version 3 supports extended range VLAN (VLANs 1006 to 4094) database propagation in VTP server mode. If extended VLANs are configured, you cannot convert from VTP version 3 to version 1 or 2.
- Before you can create a VLAN, the device must be in VTP server mode or VTP transparent mode. If the device is a VTP server, you must define a VTP domain or VTP will not function.
- The device does not support Token Ring or FDDI media. The device does not forward FDDI, FDDI-Net, TrCRE, or TrBRF traffic, but it does propagate the VLAN configuration through VTP.
- A fixed number of spanning tree instances are supported on the device (See the datasheet for the latest information). If the device has more active VLANs than the supported number of spanning tree instances, spanning tree is still enabled only on the supported number of VLANs and disabled on all remaining VLANs.

If you have already used all available spanning-tree instances on a device, adding another VLAN anywhere in the VTP domain creates a VLAN on that device that is not running spanning-tree. If you have the default allowed list on the trunk ports of that device (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent devices that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of devices that have used up their allocation of spanning-tree instances.

If the number of VLANs on the device exceeds the number of supported spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your device to map multiple VLANs to a single spanning-tree instance.

Extended-Range VLAN Configuration Guidelines

Extended-range VLANs are VLANs with IDs from 1006 to 4094.

VTP 3 only supports extended-range VLANs.

Follow these guidelines when creating extended-range VLANs:

- VLAN IDs in the extended range are not saved in the VLAN database and are not recognized by VTP unless the device is running VTP version 3.
- You cannot include extended-range VLANs in the pruning eligible range.
- For VTP version 1 or 2, you can set the VTP mode to transparent in global configuration mode. You should save this configuration to the startup configuration so that the device boots up in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the device resets. If you create extended-range VLANs in VTP version 3, you cannot convert to VTP version 1 or 2.

Default VLAN Configurations

Default Ethernet VLAN Configuration

The following table displays the default configuration for Ethernet VLANs.



Note The switch supports Ethernet interfaces exclusively. Because FDDI and Token Ring VLANs are not locally supported, you only configure FDDI and Token Ring media-specific characteristics for VTP global advertisements to other switches.

Table 209: Ethernet VLAN Defaults and Range

Parameter	Default	Range
VLAN ID	1	1 to 4094. Note Extended-range VLANs (VLAN IDs 1006 to 4094) are only saved in the VLAN database in VTP version 3.
VLAN name	VLANxxxx, where xxxx represents four numeric digits (including leading zeros) equal to the VLAN ID number	No range
IEEE 802.10 SAID	100001 (100000 plus the VLAN ID)	1 to 4294967294
IEEE 802.1Q SAID	1500	576-18190
Private VLANs	none configured	2 to 1001, 1006 to 4094

Default VLAN Configuration

You can change only the MTU size, private VLAN, and the remote SPAN configuration state on extended-range VLANs; all other characteristics must remain at the default state.



Note The switch must be running the LAN Base image to support remote SPAN.

How to Configure VLANs

How to Configure Normal-Range VLANs

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID
- VLAN name
- VLAN type
 - Ethernet
 - Fiber Distributed Data Interface [FDDI]
 - FDDI network entity title [NET]

- TrBRF or TrCRF
- Token Ring
- Token Ring-Net
- VLAN state (active or suspended)
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another

You can cause inconsistency in the VLAN database if you attempt to manually delete the *vlan.dat* file. If you want to modify the VLAN configuration, follow the procedures in this section.

Creating or Modifying an Ethernet VLAN

Each Ethernet VLAN in the VLAN database has a unique, 4-digit ID that can be a number from 1 to 1001. VLAN IDs 1002 to 1005 are reserved for Token Ring and FDDI VLANs. To create a normal-range VLAN to be added to the VLAN database, assign a number and name to the VLAN.



Note With VTP version 1 and 2, if the device is in VTP transparent mode, you can assign VLAN IDs greater than 1006, but they are not added to the VLAN database.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan *vlan-id***
4. **name *vlan-name***
5. **mtu *mtu-size***
6. **remote-span**
7. **end**
8. **show vlan {name *vlan-name* | id *vlan-id*}**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vlan <i>vlan-id</i> Example: Switch(config)# vlan 20	Enters a VLAN ID, and enters VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. Note The available VLAN ID range for this command is 1 to 4094.
Step 4	name <i>vlan-name</i> Example: Switch(config-vlan)# name test20	(Optional) Enters a name for the VLAN. If no name is entered for the VLAN, the default is to append the <i>vlan-id</i> value with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.
Step 5	mtu <i>mtu-size</i> Example: Switch(config-vlan)# mtu 256	(Optional) Changes the MTU size (or other VLAN characteristic).
Step 6	remote-span Example: Switch(config-vlan)# remote-span	(Optional) Configures the VLAN as the RSPAN VLAN for a remote SPAN session.
Step 7	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 8	show vlan {name <i>vlan-name</i> id <i>vlan-id</i>} Example: Switch# show vlan name test20 id 20	Verifies your entries.
Step 9	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Switch# <code>copy running-config startup-config</code>	

Deleting a VLAN

When you delete a VLAN from a device that is in VTP server mode, the VLAN is removed from the VLAN database for all devices in the VTP domain. When you delete a VLAN from a device that is in VTP transparent mode, the VLAN is deleted only on that specific device .

You cannot delete the default VLANs for the different media types: Ethernet VLAN 1 and FDDI or Token Ring VLANs 1002 to 1005.



Caution When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `no vlan vlan-id`
4. `end`
5. `show vlan brief`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no vlan vlan-id Example: Switch(config)# <code>no vlan 4</code>	Removes the VLAN by entering the VLAN ID.

	Command or Action	Purpose
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show vlan brief Example: Switch# show vlan brief	Verifies the VLAN removal.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Assigning Static-Access Ports to a VLAN

You can assign a static-access port to a VLAN without having VTP globally propagate VLAN configuration information by disabling VTP (VTP transparent mode).

If you assign an interface to a VLAN that does not exist, the new VLAN is created.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **switchport access vlan** *vlan-id*
5. **end**
6. **show running-config interface** *interface-id*
7. **show interfaces** *interface-id* **switchport**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example:	Enters the interface to be added to the VLAN.

	Command or Action	Purpose
	Switch(config)# interface gigabitethernet 1/0/1	
Step 3	switchport mode access Example: Switch(config-if)# switchport mode access	Defines the VLAN membership mode for the port (Layer 2 access port).
Step 4	switchport access vlan <i>vlan-id</i> Example: Switch(config-if)# switchport access vlan 2	Assigns the port to a VLAN. Valid VLAN IDs are 1 to 4094.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config interface <i>interface-id</i> Example: Switch# show running-config interface gigabitethernet 1/0/1	Verifies the VLAN membership mode of the interface.
Step 7	show interfaces <i>interface-id</i> switchport Example: Switch# show interfaces gigabitethernet 1/0/1 switchport	Verifies your entries in the <i>Administrative Mode</i> and the <i>Access Mode VLAN</i> fields of the display.

How to Configure Extended-Range VLANs

With VTP version 1 and version 2, when the switch is in VTP transparent mode (VTP disabled), you can create extended-range VLANs (in the range 1006 to 4094). VTP version supports extended-range VLANs in server or transparent mode. Extended-range VLANs enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are allowed for any **switchport** commands that allow VLAN IDs.

With VTP version 1 or 2, extended-range VLAN configurations are not stored in the VLAN database, but because VTP mode is transparent, they are stored in the switch running configuration file, and you can save the configuration in the startup configuration file by using the **copy running-config startup-config** privileged EXEC command. Extended-range VLANs created in VTP version 3 are stored in the VLAN database.

Creating an Extended-Range VLAN

You create an extended-range VLAN in global configuration mode by entering the **vlan** global configuration command with a VLAN ID from 1006 to 4094. The extended-range VLAN has the default Ethernet VLAN characteristics and the MTU size, and RSPAN configuration are the only parameters you can change. See the description of the **vlan** global configuration command in the command reference for the default settings of all parameters. In VTP version 1 or 2, if you enter an extended-range VLAN ID when the switch is not in VTP transparent mode, an error message is generated when you exit VLAN configuration mode, and the extended-range VLAN is not created.

In VTP version 1 and 2, extended-range VLANs are not saved in the VLAN database; they are saved in the switch running configuration file. You can save the extended-range VLAN configuration in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command. VTP version 3 saves extended-range VLANs in the VLAN database.

SUMMARY STEPS

1. **configure terminal**
2. **vtp mode transparent**
3. **vlan *vlan-id***
4. **mtu *mtu size***
5. **remote-span**
6. **end**
7. **show vlan id *vlan-id***
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	vtp mode transparent Example: Switch(config)# vtp mode transparent	Configures the device for VTP transparent mode, disabling VTP. Note This step is not required for VTP version 3.
Step 3	vlan <i>vlan-id</i> Example: Switch(config)# vlan 2000 Switch(config-vlan)#	Enters an extended-range VLAN ID and enters VLAN configuration mode. The range is 1006 to 4094.
Step 4	mtu <i>mtu size</i> Example:	Modifies the VLAN by changing the MTU size.

	Command or Action	Purpose
	<code>Switch(config-vlan)# mtu 1024</code>	
Step 5	remote-span Example: <code>Switch(config-vlan)# remote-span</code>	(Optional) Configures the VLAN as the RSPAN VLAN.
Step 6	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.
Step 7	show vlan id <i>vlan-id</i> Example: <code>Switch# show vlan id 2000</code>	Verifies that the VLAN has been created.
Step 8	copy running-config startup config Example: <code>Switch# copy running-config startup-config</code>	<p>Saves your entries in the device startup configuration file.</p> <p>To save an extended-range VLAN configuration, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the device startup configuration file. Otherwise, if the device resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved.</p> <p>Note This step is not required for VTP version 3 because VLANs are saved in the VLAN database.</p>

Monitoring VLANs

Table 210: Privileged EXEC show Commands

Command	Purpose
<code>show interfaces [vlan <i>vlan-id</i>]</code>	Displays characteristics for all interfaces or for the specified VLAN configured on the device.

Configuration Examples

Example: Creating a VLAN Name

This example shows how to create Ethernet VLAN 20, name it test20, and add it to the VLAN database:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# name test20
Switch(config-vlan)# end
```

Example: Configuring a Port as Access Port

This example shows how to configure a port as an access port in VLAN 2:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# end
```

Example: Creating an Extended-Range VLAN

This example shows how to create a new extended-range VLAN with all default characteristics, enter VLAN configuration mode, and save the new VLAN in the switch startup configuration file:

```
Switch(config)# vtp mode transparent
Switch(config)# vlan 2000
Switch(config-vlan)# end
Switch# copy running-config startup config
```

Where to Go Next

After configuring VLANs, you can configure the following:

- VLAN Trunking Protocol (VTP)
- VLAN trunks
- Private VLANs



CHAPTER 102

Configuring VLAN Trunks

- [Finding Feature Information, on page 2099](#)
- [Prerequisites for VLAN Trunks, on page 2099](#)
- [Information About VLAN Trunks, on page 2100](#)
- [How to Configure VLAN Trunks, on page 2103](#)
- [Configuration Examples for VLAN Trunking, on page 2116](#)
- [Where to Go Next, on page 2116](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for VLAN Trunks

The IEEE 802.1Q trunks impose these limitations on the trunking strategy for a network:

- In a network of Cisco devices connected through IEEE 802.1Q trunks, the devices maintain one spanning-tree instance for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

When you connect a Cisco device to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco device combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q device. However, spanning-tree information for each VLAN is maintained by Cisco devices separated by a cloud of non-Cisco IEEE 802.1Q devices. The non-Cisco IEEE 802.1Q cloud separating the Cisco devices is treated as a single trunk link between the devices.

- Make sure the native VLAN for an IEEE 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result.

- Disabling spanning tree on the native VLAN of an IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure your network is loop-free before disabling spanning tree.

Information About VLAN Trunks

Trunking Overview

A trunk is a point-to-point link between one or more Ethernet device interfaces and another networking device such as a router or a device. Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network.



Note You can configure a trunk on a single Ethernet interface or on an EtherChannel bundle.

Trunking Modes

Ethernet trunk interfaces support different trunking modes. You can set an interface as trunking or nontrunking or to negotiate trunking with the neighboring interface. To autonegotiate trunking, the interfaces must be in the same VTP domain.

Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP), which is a Point-to-Point Protocol (PPP). However, some internetworking devices might forward DTP frames improperly, which could cause misconfigurations.

Layer 2 Interface Modes

Table 211: Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface regardless of whether or not the neighboring interface is a trunk interface.
switchport mode dynamic auto	Makes the interface able to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to trunk or desirable mode. The default switchport mode for all Ethernet interfaces is dynamic auto .
switchport mode dynamic desirable	Makes the interface actively attempt to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to trunk , desirable , or auto mode.

Mode	Function
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport nonegotiate	Prevents the interface from generating DTP frames. You can use this command only when the interface switchport mode is access or trunk . You must manually configure the neighboring interface as a trunk interface to establish a trunk link.
switchport mode private-vlan	Configures the private VLAN mode. Note The switchport mode private-vlan command option is not supported.

Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4094, are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), DTP, and VTP in VLAN 1.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port will be added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same is true for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled, if VTP knows of the VLAN, and if the VLAN is in the allowed list for the port. When VTP detects a newly enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of the enabled VLAN. When VTP detects a new VLAN and the VLAN is not in the allowed list for a trunk port, the trunk port does not become a member of the new VLAN.

Load Sharing on Trunk Ports

Load sharing divides the bandwidth supplied by parallel trunks connecting devices. To avoid loops, STP normally blocks all but one parallel link between devices. Using load sharing, you divide the traffic between the links according to which VLAN the traffic belongs.

You configure load sharing on trunk ports by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same device. For load sharing using STP path costs, each load-sharing link can be connected to the same device or to two different devices.

Network Load Sharing Using STP Priorities

When two ports on the same device form a loop, the device uses the STP port priority to decide which port is enabled and which port is in a blocking state. You can set the priorities on a parallel trunk port so that the port carries all the traffic for a given VLAN. The trunk port with the higher priority (lower values) for a VLAN

is forwarding traffic for that VLAN. The trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

Network Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs, blocking different ports for different VLANs. The VLANs keep the traffic separate and maintain redundancy in the event of a lost link.

Feature Interactions

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the device propagates the setting that you entered to all ports in the group:
 - Allowed-VLAN list.
 - STP port priority for each VLAN.
 - STP Port Fast setting.
 - Trunk status:
 - If one port in a port group ceases to be a trunk, all ports cease to be trunks.
- We recommend that you configure no more than 24 trunk ports in Per VLAN Spanning Tree (PVST) mode and no more than 40 trunk ports in Multiple Spanning Tree (MST) mode.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.
- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x on a dynamic port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.

Default Layer 2 Ethernet Interface VLAN Configuration

The following table shows the default Layer 2 Ethernet interface VLAN configuration.

Table 212: Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Interface mode	switchport mode dynamic auto
Allowed VLAN range	VLANs 1 to 4094
VLAN range eligible for pruning	VLANs 2 to 1001

Feature	Default Setting
Default VLAN (for access ports)	VLAN 1
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1

How to Configure VLAN Trunks

To avoid trunking misconfigurations, configure interfaces connected to devices that do not support DTP to not forward DTP frames, that is, to turn off DTP.

- If you do not intend to trunk across those links, use the **switchport mode access** interface configuration command to disable trunking.
- To enable trunking to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration commands to cause the interface to become a trunk but to not generate DTP frames.

Configuring an Ethernet Interface as a Trunk Port

Configuring a Trunk Port

Because trunk ports send and receive VTP advertisements, to use VTP you must ensure that at least one trunk port is configured on the device and that this trunk port is connected to the trunk port of a second device. Otherwise, the device cannot receive any VTP advertisements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode {dynamic {auto | desirable} | trunk}**
5. **switchport access vlan *vlan-id***
6. **switchport trunk native vlan *vlan-id***
7. **end**
8. **show interfaces *interface-id* switchport**
9. **show interfaces *interface-id* trunk**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/2</code>	Specifies the port to be configured for trunking, and enters interface configuration mode.
Step 4	switchport mode {dynamic {auto desirable} trunk} Example: Switch(config-if)# <code>switchport mode dynamic desirable</code>	Configures the interface as a Layer 2 trunk (required only if the interface is a Layer 2 access port or tunnel port or to specify the trunking mode). <ul style="list-style-type: none"> • dynamic auto—Sets the interface to a trunk link if the neighboring interface is set to trunk or desirable mode. This is the default. • dynamic desirable—Sets the interface to a trunk link if the neighboring interface is set to trunk, desirable, or auto mode. • trunk—Sets the interface in permanent trunking mode and negotiate to convert the link to a trunk link even if the neighboring interface is not a trunk interface.
Step 5	switchport access vlan <i>vlan-id</i> Example: Switch(config-if)# <code>switchport access vlan 200</code>	(Optional) Specifies the default VLAN, which is used if the interface stops trunking.
Step 6	switchport trunk native vlan <i>vlan-id</i> Example: Switch(config-if)# <code>switchport trunk native vlan 200</code>	Specifies the native VLAN for IEEE 802.1Q trunks.
Step 7	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	show interfaces <i>interface-id</i> switchport Example: <pre>Switch# show interfaces gigabitethernet 1/0/2 switchport</pre>	Displays the switch port configuration of the interface in the <i>Administrative Mode</i> and the <i>Administrative Trunking Encapsulation</i> fields of the display.
Step 9	show interfaces <i>interface-id</i> trunk Example: <pre>Switch# show interfaces gigabitethernet 1/0/2 trunk</pre>	Displays the trunk configuration of the interface.
Step 10	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Defining the Allowed VLANs on a Trunk

VLAN 1 is the default VLAN on all trunk ports in all Cisco devices, and it has previously been a requirement that VLAN 1 always be enabled on every trunk link. You can use the VLAN 1 minimization feature to disable VLAN 1 on any individual VLAN trunk link so that no user traffic (including spanning-tree advertisements) is sent or received on VLAN 1.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode trunk**
5. **end**
6. **show interfaces *interface-id* switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the port to be configured, and enters interface configuration mode.
Step 4	switchport mode trunk Example: Switch(config-if)# <code>switchport mode trunk</code>	Configures the interface as a VLAN trunk port.
Step 5	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: Switch# <code>show interfaces gigabitethernet 1/0/1 switchport</code>	Verifies your entries in the <i>Trunking VLANs Enabled</i> field of the display.
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Changing the Pruning-Eligible List

The pruning-eligible list applies only to trunk ports. Each trunk port has its own eligibility list. VTP pruning must be enabled for this procedure to take effect.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `switchport trunk pruning vlan {add | except | none | remove} vlan-list [,vlan [,vlan [,],]]`

5. `end`
6. `show interfaces interface-id switchport`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>interface interface-id</code></p> <p>Example:</p> <pre>Switch(config)# interface gigabitethernet0/1</pre>	<p>Selects the trunk port for which VLANs should be pruned, and enters interface configuration mode.</p>
Step 4	<p><code>switchport trunk pruning vlan {add except none remove} vlan-list [,vlan [,vlan [,...]]</code></p>	<p>Configures the list of VLANs allowed to be pruned from the trunk.</p> <p>For explanations about using the add, except, none, and remove keywords, see the command reference for this release.</p> <p>Separate non-consecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs. Valid IDs are 2 to 1001. Extended-range VLANs (VLAN IDs 1006 to 4094) cannot be pruned.</p> <p>VLANs that are pruning-ineligible receive flooded traffic.</p> <p>The default list of VLANs allowed to be pruned contains VLANs 2 to 1001.</p>
Step 5	<p><code>end</code></p> <p>Example:</p> <pre>Switch(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p><code>show interfaces interface-id switchport</code></p> <p>Example:</p> <pre>Switch# show interfaces gigabitethernet 1/0/1</pre>	<p>Verifies your entries in the <i>Pruning VLANs Enabled</i> field of the display.</p>

	Command or Action	Purpose
	<code>switchport</code>	
Step 7	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the device forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

The native VLAN can be assigned any VLAN ID.

If a packet has a VLAN ID that is the same as the outgoing port native VLAN ID, the packet is sent untagged; otherwise, the device sends the packet with a tag.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport trunk native vlan *vlan-id***
5. **end**
6. **show interfaces *interface-id* switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example:	Defines the interface that is configured as the IEEE 802.1Q trunk, and enters interface configuration mode.

	Command or Action	Purpose
	<code>Switch(config)# interface gigabitethernet 1/0/2</code>	
Step 4	switchport trunk native vlan <i>vlan-id</i> Example: <code>Switch(config-if)# switchport trunk native vlan 12</code>	Configures the VLAN that is sending and receiving untagged traffic on the trunk port. For <i>vlan-id</i> , the range is 1 to 4094.
Step 5	end Example: <code>Switch(config-if)# end</code>	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: <code>Switch# show interfaces gigabitethernet 1/0/2 switchport</code>	Verifies your entries in the <i>Trunking Native Mode VLAN</i> field.
Step 7	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Trunk Ports for Load Sharing

Configuring Load Sharing Using STP Port Priorities

These steps describe how to configure a network with load sharing using STP port priorities.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `vtp domain domain-name`
4. `vtp mode server`
5. `end`
6. `show vtp status`
7. `show vlan`
8. `configure terminal`
9. `interface interface-id`
10. `switchport mode trunk`

11. **end**
12. **show interfaces** *interface-id* **switchport**
13. Repeat the above steps on Device A for a second port in the device.
14. Repeat the above steps on Device B to configure the trunk ports that connect to the trunk ports configured on Device A.
15. **show vlan**
16. **configure terminal**
17. **interface** *interface-id*
18. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
19. **exit**
20. **interface** *interface-id*
21. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
22. **end**
23. **show running-config**
24. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode on Device A.
Step 3	vtp domain <i>domain-name</i> Example: Switch(config)# vtp domain <i>workdomain</i>	Configures a VTP administrative domain. The domain name can be 1 to 32 characters.
Step 4	vtp mode server Example: Switch(config)# vtp mode server	Configures Device A as the VTP server.
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show vtp status Example: Switch# <code>show vtp status</code>	Verifies the VTP configuration on both Device A and Device B. In the display, check the <i>VTP Operating Mode</i> and the <i>VTP Domain Name</i> fields.
Step 7	show vlan Example: Switch# <code>show vlan</code>	Verifies that the VLANs exist in the database on Device A.
Step 8	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 9	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet1/0/1</code>	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 10	switchport mode trunk Example: Switch(config-if)# <code>switchport mode trunk</code>	Configures the port as a trunk port.
Step 11	end Example: Switch(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 12	show interfaces <i>interface-id</i> switchport Example: Switch# <code>show interfaces gigabitethernet 1/0/1 switchport</code>	Verifies the VLAN configuration.
Step 13	Repeat the above steps on Device A for a second port in the device.	
Step 14	Repeat the above steps on Device B to configure the trunk ports that connect to the trunk ports configured on Device A.	

	Command or Action	Purpose
Step 15	show vlan Example: Switch# <code>show vlan</code>	When the trunk links come up, VTP passes the VTP and VLAN information to Device B. This command verifies that Device B has learned the VLAN configuration.
Step 16	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode on Device A.
Step 17	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Defines the interface to set the STP port priority, and enters interface configuration mode.
Step 18	spanning-tree vlan <i>vlan-range</i> port-priority <i>priority-value</i> Example: Switch(config-if)# <code>spanning-tree vlan 8-10 port-priority 16</code>	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 19	exit Example: Switch(config-if)# <code>exit</code>	Returns to global configuration mode.
Step 20	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/2</code>	Defines the interface to set the STP port priority, and enters interface configuration mode.
Step 21	spanning-tree vlan <i>vlan-range</i> port-priority <i>priority-value</i> Example: Switch(config-if)# <code>spanning-tree vlan 3-6 port-priority 16</code>	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 22	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config-if) # end	
Step 23	show running-config Example: Switch# show running-config	Verifies your entries.
Step 24	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Load Sharing Using STP Path Cost

These steps describe how to configure a network with load sharing using STP path costs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode trunk**
5. **exit**
6. Repeat Steps 2 through 4 on a second interface in Device A .
7. **end**
8. **show running-config**
9. **show vlan**
10. **configure terminal**
11. **interface *interface-id***
12. **spanning-tree vlan *vlan-range* cost *cost-value***
13. **end**
14. Repeat Steps 9 through 13 on the other configured trunk interface on Device A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
15. **exit**
16. **show running-config**
17. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	<code>Switch> enable</code>	
Step 2	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode on Device A.
Step 3	interface <i>interface-id</i> Example: <code>Switch(config)# interface gigabitethernet 1/0/1</code>	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 4	switchport mode trunk Example: <code>Switch(config-if)# switchport mode trunk</code>	Configures the port as a trunk port.
Step 5	exit Example: <code>Switch(config-if)# exit</code>	Returns to global configuration mode.
Step 6	Repeat Steps 2 through 4 on a second interface in Device A.	
Step 7	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.
Step 8	show running-config Example: <code>Switch# show running-config</code>	Verifies your entries. In the display, make sure that the interfaces are configured as trunk ports.
Step 9	show vlan Example: <code>Switch# show vlan</code>	When the trunk links come up, Device A receives the VTP information from the other devices. This command verifies that Device A has learned the VLAN configuration.

	Command or Action	Purpose
Step 10	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 11	interface <i>interface-id</i> Example: Switch(config)# <code>interface gigabitethernet 1/0/1</code>	Defines the interface on which to set the STP cost, and enters interface configuration mode.
Step 12	spanning-tree vlan <i>vlan-range</i> cost <i>cost-value</i> Example: Switch(config-if)# <code>spanning-tree vlan 2-4 cost 30</code>	Sets the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 13	end Example: Switch(config-if)# <code>end</code>	Returns to global configuration mode.
Step 14	Repeat Steps 9 through 13 on the other configured trunk interface on Device A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.	
Step 15	exit Example: Switch(config)# <code>exit</code>	Returns to privileged EXEC mode.
Step 16	show running-config Example: Switch# <code>show running-config</code>	Verifies your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.
Step 17	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuration Examples for VLAN Trunking

Example: Configuring a Trunk Port

The following example shows how to configure a port as an IEEE 802.1Q trunk. The example assumes that the neighbor interface is configured to support IEEE 802.1Q trunking.

```
Switch# configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
Switch(config)# interface gigabitethernet1/0/2  
Switch(config-if)# switchport mode dynamic desirable  
Switch(config-if)# end
```

Example: Removing a VLAN from a Port

This example shows how to remove VLAN 2 from the allowed VLAN list on a port:

```
Switch(config)# interface gigabitethernet 1/0/1  
Switch(config-if)# switchport trunk allowed vlan remove 2  
Switch(config-if)# end
```

Where to Go Next

After configuring VLAN trunks, you can configure the following:

- VLANs
- Private VLANs



CHAPTER 103

Configuring VMPS

- [Finding Feature Information, on page 2117](#)
- [Prerequisites for VMPS, on page 2117](#)
- [Restrictions for VMPS, on page 2117](#)
- [Information About VMPS, on page 2118](#)
- [How to Configure VMPS, on page 2120](#)
- [Monitoring the VMPS, on page 2126](#)
- [Configuration Example for VMPS, on page 2127](#)
- [Where to Go Next, on page 2128](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for VMPS

You should configure the VLAN Membership Policy Server (VMPS) before you configure ports as dynamic-access ports.

When you configure a port as a dynamic-access port, the spanning-tree Port Fast feature is automatically enabled for that port. The Port Fast mode accelerates the process of bringing the port into the forwarding state.

The VTP management domain of the VMPS client and the VMPS server must be the same.

Restrictions for VMPS

The following are restrictions for configuring VMPS:

- IEEE 802.1x ports cannot be configured as dynamic-access ports. If you try to enable IEEE 802.1x on a dynamic-access (VQP) port, an error message appears, and IEEE 802.1x is not enabled. If you try to change an IEEE 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
- Trunk ports cannot be dynamic-access ports, but you can enter the **switchport access vlan dynamic** interface configuration command for a trunk port. In this case, the device retains the setting and applies it if the port is later configured as an access port. You must turn off trunking on the port before the dynamic-access setting takes effect.
- Dynamic-access ports cannot be monitor ports.
- Secure ports cannot be dynamic-access ports. You must disable port security on a port before it becomes dynamic.
- Private VLAN ports cannot be dynamic-access ports
- Dynamic-access ports cannot be members of an EtherChannel group.
- Port channels cannot be configured as dynamic-access ports.
- The VLAN configured on the VMPS server should not be a voice VLAN.

Information About VMPS

Dynamic VLAN Assignments

The VLAN Query Protocol (VQP) is used to support dynamic-access ports, which are not permanently assigned to a VLAN, but give VLAN assignments based on the MAC source addresses seen on the port. Each time an unknown MAC address is seen, the device sends a VQP query to a remote VLAN Membership Policy Server (VMPS); the query includes the newly seen MAC address and the port on which it was seen. The VMPS responds with a VLAN assignment for the port. The device cannot be a VMPS server but can act as a client to the VMPS and communicate with it through VQP.

Each time the client device receives the MAC address of a new host, it sends a VQP query to the VMPS. When the VMPS receives this query, it searches its database for a MAC-address-to-VLAN mapping. The server response is based on this mapping and whether or not the server is in open or secure mode. In secure mode, the server shuts down the port when an illegal host is detected. In open mode, the server denies the host access to the port.

If the port is currently unassigned (that is, it does not yet have a VLAN assignment), the VMPS provides one of these responses:

- If the host is allowed on the port, the VMPS sends the client a vlan-assignment response containing the assigned VLAN name and allowing access to the host.
- If the host is not allowed on the port and the VMPS is in open mode, the VMPS sends an access-denied response.
- If the VLAN is not allowed on the port and the VMPS is in secure mode, the VMPS sends a port-shutdown response.

If the port already has a VLAN assignment, the VMPS provides one of these responses:

- If the VLAN in the database matches the current VLAN on the port, the VMPS sends an success response, allowing access to the host.
- If the VLAN in the database does not match the current VLAN on the port and active hosts exist on the port, the VMPS sends an access-denied or a port-shutdown response, depending on the secure mode of the VMPS.

If the device receives an access-denied response from the VMPS, it continues to block traffic to and from the host MAC address. The device continues to monitor the packets directed to the port and sends a query to the VMPS when it identifies a new host address. If the device receives a port-shutdown response from the VMPS, it disables the port. The port must be manually reenabled by using Network Assistant, the CLI, or SNMP.

Dynamic-Access Port VLAN Membership

A dynamic-access port can belong to only one VLAN with an ID from 1 to 4094. When the link comes up, the device does not forward traffic to or from this port until the VMPS provides the VLAN assignment. The VMPS receives the source MAC address from the first packet of a new host connected to the dynamic-access port and attempts to match the MAC address to a VLAN in the VMPS database.

If there is a match, the VMPS sends the VLAN number for that port. If the client device was not previously configured, it uses the domain name from the first VTP packet it receives on its trunk port from the VMPS. If the client device was previously configured, it includes its domain name in the query packet to the VMPS to obtain its VLAN number. The VMPS verifies that the domain name in the packet matches its own domain name before accepting the request and responds to the client with the assigned VLAN number for the client. If there is no match, the VMPS either denies the request or shuts down the port (depending on the VMPS secure mode setting).

Multiple hosts (MAC addresses) can be active on a dynamic-access port if they are all in the same VLAN; however, the VMPS shuts down a dynamic-access port if more than 20 hosts are active on the port.

If the link goes down on a dynamic-access port, the port returns to an isolated state and does not belong to a VLAN. Any hosts that come online through the port are checked again through the VQP with the VMPS before the port is assigned to a VLAN.

Dynamic-access ports can be used for direct host connections, or they can connect to a network. A maximum of 20 MAC addresses are allowed per port on the device. A dynamic-access port can belong to only one VLAN at a time, but the VLAN can change over time, depending on the MAC addresses seen.

Default VMPS Client Configuration

The following table shows the default VMPS and dynamic-access port configuration on client switches.

Table 213: Default VMPS Client and Dynamic-Access Port Configuration

Feature	Default Setting
VMPS domain server	None
VMPS reconfirm interval	60 minutes
VMPS server retry count	3
Dynamic-access ports	None configured

How to Configure VMPS

Entering the IP Address of the VMPS



Note If the VMPS is being defined for a cluster of switches, enter the address on the command switch.

Before you begin

You must first enter the IP address of the server to configure the switch as a client.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vmpls server *ipaddress* primary**
4. **vmpls server *ipaddress***
5. **end**
6. **show vmpls**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vmpls server <i>ipaddress</i> primary Example: Switch(config)# vmpls server 10.1.2.3 primary	Enters the IP address of the device acting as the primary VMPS server.
Step 4	vmpls server <i>ipaddress</i> Example:	(Optional) Enters the IP address of the device acting as a secondary VMPS server. You can enter up to three secondary server addresses.

	Command or Action	Purpose
	<code>Switch(config)# vmps server 10.3.4.5</code>	
Step 5	end Example: <code>Switch(config)# end</code>	Returns to privileged EXEC mode.
Step 6	show vmps Example: <code>Switch# show vmps</code>	Verifies your entries in the <i>VMPS Domain Server</i> field of the display.
Step 7	copy running-config startup-config Example: <code>Switch# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Dynamic-Access Ports on VMPS Clients



Caution Dynamic-access port VLAN membership is for end stations or hubs connected to end stations. Connecting dynamic-access ports to other switches can cause a loss of connectivity.

If you are configuring a port on a cluster member device as a dynamic-access port, first use the **recommand** privileged EXEC command to log in to the cluster member device.

Before you begin

You must have IP connectivity to the VMPS for dynamic-access ports to work. You can test for IP connectivity by pinging the IP address of the VMPS and verifying that you get a response.



Note To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To return an interface to its default switchport mode (dynamic auto), use the **no switchport mode** interface configuration command. To reset the access mode to the default VLAN for the device, use the **no switchport access vlan** interface configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*

4. `switchport mode access`
5. `switchport access vlan dynamic`
6. `end`
7. `show interfaces interface-id switchport`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface interface-id Example: <pre>Switch(config)# interface gigabitethernet 0/1</pre>	Specifies the device port that is connected to the end station, and enters interface configuration mode.
Step 4	switchport mode access Example: <pre>Switch(config-if)# switchport mode access</pre>	Sets the port to access mode.
Step 5	switchport access vlan dynamic Example: <pre>Switch(config-if)# switchport access vlan dynamic</pre>	Configures the port as eligible for dynamic VLAN membership. The dynamic-access port must be connected to an end station.
Step 6	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show interfaces interface-id switchport Example: <pre>Switch# show interfaces gigabitethernet 0/1</pre>	Verifies your entries in the <i>Operational Mode</i> field of the display.

	Command or Action	Purpose
	<code>switchport</code>	
Step 8	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Reconfirming VLAN Memberships

This task confirms the dynamic-access port VLAN membership assignments that the device has received from the VMPS.

SUMMARY STEPS

1. `enable`
2. `vmps reconfirm`
3. `show vmps`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	vmps reconfirm Example: <pre>Switch# vmps reconfirm</pre>	Reconfirms dynamic-access port VLAN membership.
Step 3	show vmps Example: <pre>Switch# show vmps</pre>	Verifies the dynamic VLAN reconfirmation status.

Changing the Reconfirmation Interval

VMPS clients periodically reconfirm the VLAN membership information received from the VMPS. You can set the number of minutes after which reconfirmation occurs.



Note If you are configuring a member device in a cluster, this parameter must be equal to or greater than the reconfirmation setting on the command device. You also must first use the **rcommand** privileged EXEC command to log in to the member device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vmpls reconfirm** *minutes*
4. **end**
5. **show vmpls**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vmpls reconfirm <i>minutes</i> Example: Switch(config)# vmpls reconfirm 90	Sets the number of minutes between reconfirmations of the dynamic VLAN membership. The range is 1 to 120. The default is 60 minutes.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 5	show vmpls Example: Switch# show vmpls	Verifies the dynamic VLAN reconfirmation status in the <i>Reconfirm Interval</i> field of the display.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Changing the Retry Count

Follow these steps to change the number of times that the device attempts to contact the VMPS before querying the next server.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `vmps retry count`
4. `end`
5. `show vmps`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 3	vmps retry count Example: Switch(config)# <code>vmps retry 5</code>	Changes the retry count. The retry range is 1 to 10; the default is 3.
Step 4	end Example: Switch(config)# <code>end</code>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show vmmps Example: Switch# show vmmps	Verifies your entry in the <i>Server Retry Count</i> field of the display.
Step 6	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Troubleshooting Dynamic-Access Port VLAN Membership

Problem The VMPS shuts down a dynamic-access port under these conditions:

- **Problem** The VMPS is in secure mode, and it does not allow the host to connect to the port. The VMPS shuts down the port to prevent the host from connecting to the network.
- **Problem** More than 20 active hosts reside on a dynamic-access port.

Solution To reenab a disabled dynamic-access port, enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command.

Monitoring the VMPS

You can display information about the VMPS by using the **show vmmps** privileged EXEC command. The device displays this information about the VMPS:

- **VMPS VQP Version**—The version of VQP used to communicate with the VMPS. The device queries the VMPS that is using VQP Version 1.
- **Reconfirm Interval**—The number of minutes the device waits before reconfirming the VLAN-to-MAC-address assignments.
- **Server Retry Count**—The number of times VQP resends a query to the VMPS. If no response is received after this many tries, the device starts to query the secondary VMPS.
- **VMPS domain server**—The IP address of the configured VLAN membership policy servers. The device sends queries to the one marked *current*. The one marked *primary* is the primary server.
- **VMPS Action**—The result of the most recent reconfirmation attempt. A reconfirmation attempt can occur automatically when the reconfirmation interval expires, or you can force it by entering the **vmmps reconfirm** privileged EXEC command or its Network Assistant or SNMP equivalent.

This is an example of output for the **show vmmps** privileged EXEC command:

```
Switch# show vmmps
VQP Client Status:
-----
```

```
VMPS VQP Version: 1
Reconfirm Interval: 60 min
Server Retry Count: 3
VMPS domain server: 172.20.128.86 (primary, current)
                   172.20.128.87
```

```
Reconfirmation status
-----
```

```
VMPS Action:          other
```

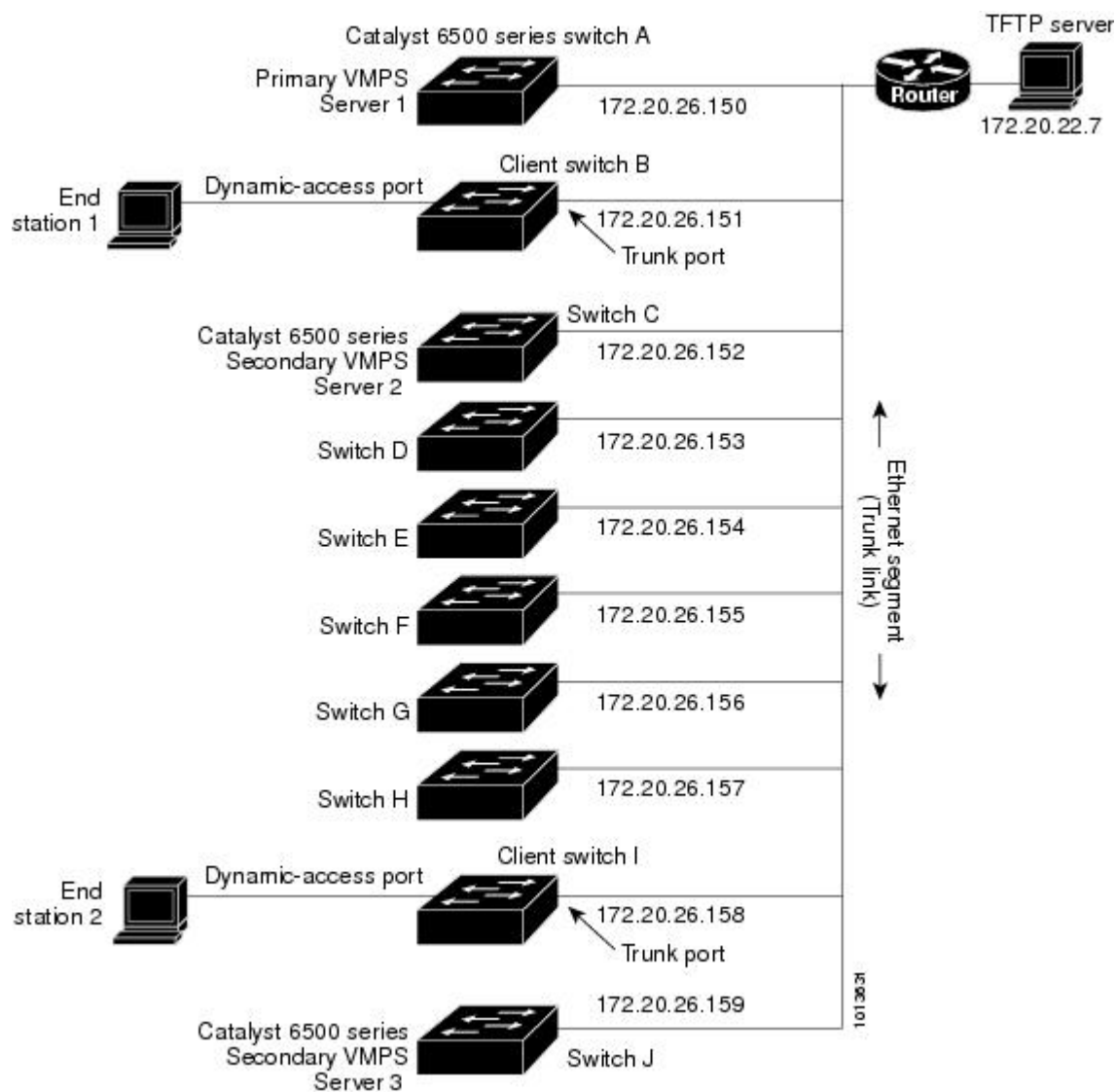
Configuration Example for VMPS

Example: VMPS Configuration

Figure 135: Dynamic Port VLAN Membership Configuration

This network has a VMPS server switch and VMPS client switches with dynamic-access ports with this configuration:

- The VMPS server and the VMPS client are separate switches.
- The Catalyst 6500 series Switch A is the primary VMPS server.
- The Catalyst 6500 series Switch C and Switch J are secondary VMPS servers.
- End stations are connected to the clients, Switch B and Switch I.
- The database configuration file is stored on the TFTP server with the IP address 172.20.22.7.



Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN Trunking
- Private VLANs
- Voice VLANs



CHAPTER 104

Configuring Voice VLANs

- [Finding Feature Information, on page 2129](#)
- [Prerequisites for Voice VLANs, on page 2129](#)
- [Restrictions for Voice VLANs, on page 2130](#)
- [Information About Voice VLAN, on page 2130](#)
- [How to Configure Voice VLAN, on page 2132](#)
- [Monitoring Voice VLAN, on page 2136](#)
- [Configuration Examples, on page 2136](#)
- [Where to Go Next, on page 2136](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Voice VLANs

The following are the prerequisites for voice VLANs:

- Voice VLAN configuration is only supported on device access ports; voice VLAN configuration is not supported on trunk ports.



Note Trunk ports can carry any number of voice VLANs, similar to regular VLANs. The configuration of voice VLANs is not supported on trunk ports.

- Before you enable voice VLAN, we recommend that you enable QoS on the device by entering the **mls qos** global configuration command and configure the port trust state to trust by entering the **mls qos trust cos** interface configuration command. If you use the auto-QoS feature, these settings are automatically configured.

- You must enable CDP on the device port connected to the Cisco IP Phone to send the configuration to the phone. (CDP is globally enabled by default on all device interfaces.)

Restrictions for Voice VLANs

You cannot configure static secure MAC addresses in the voice VLAN.

Information About Voice VLAN

Voice VLANs

The voice VLAN feature enables access ports to carry IP voice traffic from an IP phone. When the device is connected to a Cisco 7960 IP Phone, the phone sends voice traffic with Layer 3 IP precedence and Layer 2 class of service (CoS) values, which are both set to 5 by default. Because the sound quality of an IP phone call can deteriorate if the data is unevenly sent, the device supports quality of service (QoS) based on IEEE 802.1p CoS. QoS uses classification and scheduling to send network traffic from the device in a predictable manner.

The Cisco 7960 IP Phone is a configurable device, and you can configure it to forward traffic with an IEEE 802.1p priority. You can configure the device to trust or override the traffic priority assigned by a Cisco IP Phone.

Cisco IP Phone Voice Traffic

You can configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. You can configure access ports on the device to send Cisco Discovery Protocol (CDP) packets that instruct an attached phone to send voice traffic to the device in any of these ways:

- In the voice VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN, untagged (no Layer 2 CoS priority value)



Note In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5 for voice traffic and 3 for voice control traffic).

Cisco IP Phone Data Traffic

The device can also process tagged data traffic (traffic in IEEE 802.1Q or IEEE 802.1p frame types) from the device attached to the access port on the Cisco IP Phone. You can configure Layer 2 access ports on the device to send CDP packets that instruct the attached phone to configure the phone access port in one of these modes:

- In trusted mode, all traffic received through the access port on the Cisco IP Phone passes through the phone unchanged.
- In untrusted mode, all traffic in IEEE 802.1Q or IEEE 802.1p frames received through the access port on the Cisco IP Phone receive a configured Layer 2 CoS value. The default Layer 2 CoS value is 0. Untrusted mode is the default.



Note Untagged traffic from the device attached to the Cisco IP Phone passes through the phone unchanged, regardless of the trust state of the access port on the phone.

Voice VLAN Configuration Guidelines

- Because a Cisco 7960 IP Phone also supports a connection to a PC or other device, a port connecting the device to a Cisco IP Phone can carry mixed traffic. You can configure a port to decide how the Cisco IP Phone carries voice traffic and data traffic.
- The voice VLAN should be present and active on the device for the IP phone to correctly communicate on the voice VLAN. Use the **show vlan** privileged EXEC command to see if the VLAN is present (listed in the display). If the VLAN is not listed, create the voice VLAN.
- The Power over Ethernet (PoE) devices are capable of automatically providing power to Cisco pre-standard and IEEE 802.3af-compliant powered devices if they are not being powered by an AC power source.
- The Port Fast feature is automatically enabled when voice VLAN is configured. When you disable voice VLAN, the Port Fast feature is not automatically disabled.
- If the Cisco IP Phone and a device attached to the phone are in the same VLAN, they must be in the same IP subnet. These conditions indicate that they are in the same VLAN:
 - They both use IEEE 802.1p or untagged frames.
 - The Cisco IP Phone uses IEEE 802.1p frames, and the device uses untagged frames.
 - The Cisco IP Phone uses untagged frames, and the device uses IEEE 802.1p frames.
 - The Cisco IP Phone uses IEEE 802.1Q frames, and the voice VLAN is the same as the access VLAN.
- The Cisco IP Phone and a device attached to the phone cannot communicate if they are in the same VLAN and subnet but use different frame types because traffic in the same subnet is not routed (routing would eliminate the frame type difference).
- Voice VLAN ports can also be these port types:
 - Dynamic access port.
 - IEEE 802.1x authenticated port.



Note If you enable IEEE 802.1x on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the phone loses connectivity to the device for up to 30 seconds.

- Protected port.
- A source or destination port for a SPAN or RSPAN session.
- Secure port.



Note When you enable port security on an interface that is also configured with a voice VLAN, you must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN. When the port is connected to a Cisco IP Phone, the phone requires up to two MAC addresses. The phone address is learned on the voice VLAN and might also be learned on the access VLAN. Connecting a PC to the phone requires additional MAC addresses.

Default Voice VLAN Configuration

The voice VLAN feature is disabled by default.

When the voice VLAN feature is enabled, all untagged traffic is sent according to the default CoS priority of the port.

The CoS value is not trusted for IEEE 802.1p or IEEE 802.1Q tagged traffic.

How to Configure Voice VLAN

Configuring Cisco IP Phone Voice Traffic

You can configure a port connected to the Cisco IP Phone to send CDP packets to the phone to configure the way in which the phone sends voice traffic. The phone can carry voice traffic in IEEE 802.1Q frames for a specified voice VLAN with a Layer 2 CoS value. It can use IEEE 802.1p priority tagging to give voice traffic a higher priority and forward all voice traffic through the native (access) VLAN. The Cisco IP Phone can also send untagged voice traffic or use its own configuration to send voice traffic in the access VLAN. In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **mls qos trust cos**
5. **switchport voice** {vlan {*vlan-id* | dot1p | none | untagged}}
6. **end**
7. Use one of the following:
 - **show interfaces** *interface-id* **switchport**
 - **show running-config interface** *interface-id*

8. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	mls qos trust cos Example: Device(config-if)# mls qos trust cos	Configures the interface to classify incoming traffic packets by using the packet CoS value. For untagged packets, the port default CoS value is used. <p>Note Before configuring the port trust state, you must first globally enable QoS by using the mls qos global configuration command.</p>
Step 5	switchport voice {vlan{<i>vlan-id</i> dot1p none untagged}} Example: Device(config-if)# switchport voice vlan dot1p	Configures the voice VLAN. <ul style="list-style-type: none"> • vlan-id—Configures the phone to forward all voice traffic through the specified VLAN. By default, the Cisco IP Phone forwards the voice traffic with an IEEE 802.1Q priority of 5. Valid VLAN IDs are 1 to 4094. • dot1p—Configures the device to accept voice and data IEEE 802.1p priority frames tagged with VLAN ID 0 (the native VLAN). By default, the device drops all voice and data traffic tagged with VLAN 0. If configured for 802.1p the Cisco IP Phone forwards the traffic with an IEEE 802.1p priority of 5. • none—Allows the phone to use its own configuration to send untagged voice traffic. • untagged—Configures the phone to send untagged voice traffic.

	Command or Action	Purpose
Step 6	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 7	Use one of the following: <ul style="list-style-type: none"> • show interfaces <i>interface-id</i> switchport • show running-config interface <i>interface-id</i> Example: <pre>Device# show interfaces gigabitethernet 1/0/1 switchport</pre> or <pre>Device# show running-config interface gigabitethernet 1/0/1</pre>	Verifies your voice VLAN entries or your QoS and voice VLAN entries.
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the Priority of Incoming Data Frames

You can connect a PC or other data device to a Cisco IP Phone port. To process tagged data traffic (in IEEE 802.1Q or IEEE 802.1p frames), you can configure the device to send CDP packets to instruct the phone how to send data packets from the device attached to the access port on the Cisco IP Phone. The PC can generate packets with an assigned CoS value. You can configure the phone to not change (trust) or to override (not trust) the priority of frames arriving on the phone port from connected devices.

Follow these steps to set the priority of data traffic received from the non-voice port on the Cisco IP Phone:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport priority extend** {*cos value* | **trust**}
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/1</pre>	Specifies the interface connected to the Cisco IP Phone, and enters interface configuration mode.
Step 4	switchport priority extend {<i>cos value</i> trust} Example: <pre>Switch(config-if)# switchport priority extend trust</pre>	Sets the priority of data traffic received from the Cisco IP Phone access port: <ul style="list-style-type: none"> • cos value—Configures the phone to override the priority received from the PC or the attached device with the specified CoS value. The value is a number from 0 to 7, with 7 as the highest priority. The default priority is cos 0. • trust—Configures the phone access port to trust the priority received from the PC or the attached device.
Step 5	end Example: <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: <pre>Switch# show interfaces gigabitethernet1/0/1 switchport</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring Voice VLAN

To display voice VLAN configuration for an interface, use the **show interfaces *interface-id* switchport** privileged EXEC command.

Configuration Examples

Example: Configuring Cisco IP Phone Voice Traffic

This example shows how to configure a port connected to a Cisco IP Phone to use the CoS value to classify incoming traffic and to accept voice and data priority traffic tagged with VLAN ID 0:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# switchport voice vlan dot1p
Switch(config-if)# end
```

To return the port to its default setting, use the **no switchport voice vlan** interface configuration command.

Example: Configuring the Priority of Incoming Data Frames

This example shows how to configure a port connected to a Cisco IP Phone to not change the priority of frames received from the PC or the attached device:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport priority extend trust
Switch(config-if)# end
```

To return the port to its default setting, use the **no switchport priority extend** interface configuration command.

Where to Go Next

After configuring voice VLANs, you can configure the following:

- VLANs
- VLAN Trunking
- VTP
- Private VLANs



CHAPTER 105

Configuring Private VLANs

- [Finding Feature Information, on page 2137](#)
- [Prerequisites for Private VLANs, on page 2137](#)
- [Restrictions for Private VLANs, on page 2137](#)
- [Information About Private VLANs, on page 2139](#)
- [How to Configure Private VLANs, on page 2145](#)
- [Monitoring Private VLANs, on page 2154](#)
- [Configuration Examples for Private VLANs, on page 2154](#)
- [Where to Go Next, on page 2156](#)
- [Additional References, on page 2156](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Private VLANs

Private vlans are supported in transparent mode for VTP 1, 2 and 3. Private VLANs are also supported on server mode with VTP 3.

When configuring private VLANs on the device, always use the default Switch Database Management (SDM) template to balance system resources between unicast routes and Layer 2 entries. If another SDM template is configured, use the **sdm prefer default** global configuration command to set the default template.

Restrictions for Private VLANs

Private VLANs are not supported on switches running the LAN Base image.



Note In some cases, the configuration is accepted with no error messages, but the commands have no effect.

- Do not configure fallback bridging on devices with private VLANs.
- Do not configure a remote SPAN (RSPAN) VLAN as a private-VLAN primary or secondary VLAN.
- Do not configure private-VLAN ports on interfaces configured for these other features:
 - Dynamic-access port VLAN membership
 - Dynamic Trunking Protocol (DTP)
 - IPv6 Security Group (SG)
 - Port Aggregation Protocol (PAgP)
 - Link Aggregation Control Protocol (LACP)
 - Multicast VLAN Registration (MVR)
 - Voice VLAN
 - Web Cache Communication Protocol (WCCP)
- You can configure IEEE 802.1x port-based authentication on a private-VLAN port, but do not configure 802.1x with port security, voice VLAN, or per-user ACL on private-VLAN ports.
- A private-VLAN host or promiscuous port cannot be a SPAN destination port. If you configure a SPAN destination port as a private-VLAN port, the port becomes inactive.
- If you configure a static MAC address on a promiscuous port in the primary VLAN, you need not add the same static address to all associated secondary VLANs. Similarly, if you configure a static MAC address on a host port in a secondary VLAN, you need not add the same static MAC address to the associated primary VLAN. Also, when you delete a static MAC address from a private-VLAN port, you do not have to remove all instances of the configured MAC address from the private VLAN.



Note Dynamic MAC addresses learned in Secondary VLAN of a private VLAN are replicated to the Primary VLANs. All mac entries are learnt on secondary VLANs, even if the traffic ingresses from primary VLAN. If a mac-address is dynamically learnt in the primary VLAN it will not get replicated in the associated secondary VLANs.

- Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs.

Information About Private VLANs

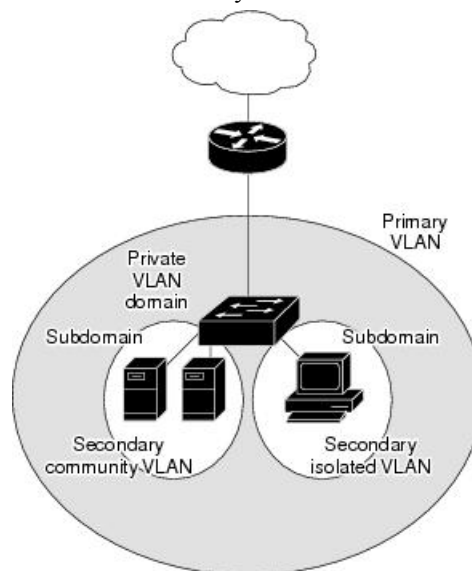
Private VLAN Domains

The private VLAN feature addresses two problems that service providers face when using VLANs:

- To enable IP routing, each VLAN is assigned a subnet address space or a block of addresses, which can result in wasting the unused IP addresses, and cause IP address management problems.

Figure 136: Private VLAN Domain

Using private VLANs addresses the scalability problem and provides IP address management benefits for service providers and Layer 2 security for customers. Private VLANs partition a regular VLAN domain into subdomains. A subdomain is represented by a pair of VLANs: a *primary* VLAN and a *secondary* VLAN. A private VLAN can have multiple VLAN pairs, one pair for each subdomain. All VLAN pairs in a private VLAN share the same primary VLAN. The secondary VLAN ID differentiates one subdomain from another.



Secondary VLANs

There are two types of secondary VLANs:

- Isolated VLANs—Ports within an isolated VLAN cannot communicate with each other at the Layer 2 level.
- Community VLANs—Ports within a community VLAN can communicate with each other but cannot communicate with ports in other communities at the Layer 2 level.

Private VLANs Ports

Private VLANs provide Layer 2 isolation between ports within the same private VLAN. Private VLAN ports are access ports that are one of these types:

- **Promiscuous**—A promiscuous port belongs to the primary VLAN and can communicate with all interfaces, including the community and isolated host ports that belong to the secondary VLANs associated with the primary VLAN.
- **Isolated**—An isolated port is a host port that belongs to an isolated secondary VLAN. It has complete Layer 2 separation from other ports within the same private VLAN, except for the promiscuous ports. Private VLANs block all traffic to isolated ports except traffic from promiscuous ports. Traffic received from an isolated port is forwarded only to promiscuous ports.
- **Community**—A community port is a host port that belongs to a community secondary VLAN. Community ports communicate with other ports in the same community VLAN and with promiscuous ports. These interfaces are isolated at Layer 2 from all other interfaces in other communities and from isolated ports within their private VLAN.



Note Trunk ports carry traffic from regular VLANs and also from primary, isolated, and community VLANs.

Primary and secondary VLANs have these characteristics:

- **Primary VLAN**—A private VLAN has only one primary VLAN. Every port in a private VLAN is a member of the primary VLAN. The primary VLAN carries unidirectional traffic downstream from the promiscuous ports to the (isolated and community) host ports and to other promiscuous ports.
- **Isolated VLAN**—A private VLAN has only one isolated VLAN. An isolated VLAN is a secondary VLAN that carries unidirectional traffic upstream from the hosts toward the promiscuous ports and the gateway.
- **Community VLAN**—A community VLAN is a secondary VLAN that carries upstream traffic from the community ports to the promiscuous port gateways and to other host ports in the same community. You can configure multiple community VLANs in a private VLAN.

A promiscuous port can serve only one primary VLAN, one isolated VLAN, and multiple community VLANs. Layer 3 gateways are typically connected to the device through a promiscuous port. With a promiscuous port, you can connect a wide range of devices as access points to a private VLAN. For example, you can use a promiscuous port to monitor or back up all the private VLAN servers from an administration workstation.

Private VLANs in Networks

In a switched environment, you can assign an individual private VLAN and associated IP subnet to each individual or common group of end stations. The end stations need to communicate only with a default gateway to communicate outside the private VLAN.

You can use private VLANs to control access to end stations in these ways:

- Configure selected interfaces connected to end stations as isolated ports to prevent any communication at Layer 2. For example, if the end stations are servers, this configuration prevents Layer 2 communication between the servers.
- Configure interfaces connected to default gateways and selected end stations (for example, backup servers) as promiscuous ports to allow all end stations access to a default gateway.

You can extend private VLANs across multiple devices by trunking the primary, isolated, and community VLANs to other devices that support private VLANs. To maintain the security of your private VLAN

configuration and to avoid other use of the VLANs configured as private VLANs, configure private VLANs on all intermediate devices, including devices that have no private VLAN ports.

IP Addressing Scheme with Private VLANs

Assigning a separate VLAN to each customer creates an inefficient IP addressing scheme:

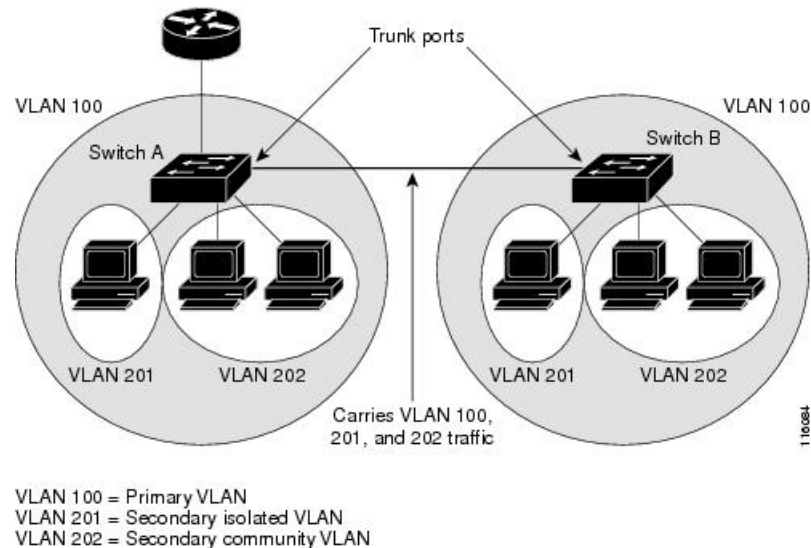
- Assigning a block of addresses to a customer VLAN can result in unused IP addresses.
- If the number of devices in the VLAN increases, the number of assigned address might not be large enough to accommodate them.

These problems are reduced by using private VLANs, where all members in the private VLAN share a common address space, which is allocated to the primary VLAN. Hosts are connected to secondary VLANs, and the DHCP server assigns them IP addresses from the block of addresses allocated to the primary VLAN. Subsequent IP addresses can be assigned to customer devices in different secondary VLANs, but in the same primary VLAN. When new devices are added, the DHCP server assigns them the next available address from a large pool of subnet addresses.

Private VLANs Across Multiple Devices

Figure 137: Private VLANs Across Switches

As with regular VLANs, private VLANs can span multiple devices. A trunk port carries the primary VLAN and secondary VLANs to a neighboring device. The trunk port treats the private VLAN as any other VLAN. A feature of private VLANs across multiple devices is that traffic from an isolated port in Device A does not reach an isolated port on Device B.



Private VLANs are supported in transparent mode for VTP 1, 2 and 3. Private vlan is also supported on server mode for VTP 3. If we have a server client setup using VTP 3, private vlans configured on the server should be reflected on the client.

Private-VLAN Interaction with Other Features

Private VLANs and Unicast, Broadcast, and Multicast Traffic

In regular VLANs, devices in the same VLAN can communicate with each other at the Layer 2 level, but devices connected to interfaces in different VLANs must communicate at the Layer 3 level. In private VLANs, the promiscuous ports are members of the primary VLAN, while the host ports belong to secondary VLANs. Because the secondary VLAN is associated to the primary VLAN, members of these VLANs can communicate with each other at the Layer 2 level.

In a regular VLAN, broadcasts are forwarded to all ports in that VLAN. Private VLAN broadcast forwarding depends on the port sending the broadcast:

- An isolated port sends a broadcast only to the promiscuous ports or trunk ports.
- A community port sends a broadcast to all promiscuous ports, trunk ports, and ports in the same community VLAN.
- A promiscuous port sends a broadcast to all ports in the private VLAN (other promiscuous ports, trunk ports, isolated ports, and community ports).

Multicast traffic is routed or bridged across private VLAN boundaries and within a single community VLAN. Multicast traffic is not forwarded between ports in the same isolated VLAN or between ports in different secondary VLANs.

Private VLAN multicast forwarding supports the following:

- Sender can be outside the VLAN and the Receivers can be inside the VLAN domain.
- Sender can be inside the VLAN and the Receivers can be outside the VLAN domain.
- Sender and Receiver can both be in the same community vlan.

Private VLANs and SVIs

In a Layer 3 device, a device virtual interface (SVI) represents the Layer 3 interface of a VLAN. Layer 3 devices communicate with a private VLAN only through the primary VLAN and not through secondary VLANs. Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs. You cannot configure Layer 3 VLAN interfaces for secondary VLANs. SVIs for secondary VLANs are inactive while the VLAN is configured as a secondary VLAN.

- If you try to configure a VLAN with an active SVI as a secondary VLAN, the configuration is not allowed until you disable the SVI.
- If you try to create an SVI on a VLAN that is configured as a secondary VLAN and the secondary VLAN is already mapped at Layer 3, the SVI is not created, and an error is returned. If the SVI is not mapped at Layer 3, the SVI is created, but it is automatically shut down.

When the primary VLAN is associated with and mapped to the secondary VLAN, any configuration on the primary VLAN is propagated to the secondary VLAN SVIs. For example, if you assign an IP subnet to the primary VLAN SVI, this subnet is the IP subnet address of the entire private VLAN.

Private-VLAN Configuration Guidelines

Secondary and Primary VLAN Configuration

Follow these guidelines when configuring private VLANs:

- Private VLANs are supported in transparent mode for VTP 1, 2 and 3. If the device is running VTP version 1 or 2, you must set VTP to transparent mode. After you configure a private VLAN, you should not change the VTP mode to client or server. VTP version 3 supports private VLANs in all modes.
- With VTP version 1 or 2, after you have configured private VLANs, use the **copy running-config startup-config** privileged EXEC command to save the VTP transparent mode configuration and private-VLAN configuration in the device startup configuration file. Otherwise, if the device resets, it defaults to VTP server mode, which does not support private VLANs. VTP version 3 does support private VLANs.
- VTP version 1 and 2 do not propagate private-VLAN configuration. You must configure private VLANs on each device where you want private-VLAN ports unless the devices are running VTP version 3, as VTP3 propagate private vlans.
- You cannot configure VLAN 1 or VLANs 1002 to 1005 as primary or secondary VLANs. Extended VLANs (VLAN IDs 1006 to 4094) can belong to private VLANs.
- A primary VLAN can have one isolated VLAN and multiple community VLANs associated with it. An isolated or community VLAN can have only one primary VLAN associated with it.
- Although a private VLAN contains more than one VLAN, only one Spanning Tree Protocol (STP) instance runs for the entire private VLAN. When a secondary VLAN is associated with the primary VLAN, the STP parameters of the primary VLAN are propagated to the secondary VLAN.
- When copying a PVLAN configuration from a tftp server and applying it on a running-config, the PVLAN association will not be formed. You will need to check and ensure that the primary VLAN is associated to all the secondary VLANs.

You can also use **configure replace flash:config_file force** instead of **copy flash:config_file running-config**.

- You can enable DHCP snooping on private VLANs. When you enable DHCP snooping on the primary VLAN, it is propagated to the secondary VLANs. If you configure DHCP on a secondary VLAN, the configuration does not take effect if the primary VLAN is already configured.
- When you enable IP source guard on private-VLAN ports, you must enable DHCP snooping on the primary VLAN.
- We recommend that you prune the private VLANs from the trunks on devices that carry no traffic in the private VLANs.
- You can apply different quality of service (QoS) configurations to primary, isolated, and community VLANs.
- Note the following considerations for sticky ARP:
 - Sticky ARP entries are those learned on SVIs and Layer 3 interfaces. These entries do not age out.
 - The **ip sticky-arp** global configuration command is supported only on SVIs belonging to private VLANs.
 - The **ip sticky-arp** interface configuration command is only supported on:

- Layer 3 interfaces
- SVIs belonging to normal VLANs
- SVIs belonging to private VLANs

For more information about using the **ip sticky-arp global** configuration and the **ip sticky-arp interface** configuration commands, see the command reference for this release.

- You can configure VLAN maps on primary and secondary VLANs. However, we recommend that you configure the same VLAN maps on private-VLAN primary and secondary VLANs.
- PVLANS are bidirectional. They can be applied at both the ingress and egress sides.

When a frame in Layer-2 is forwarded within a private VLAN, the VLAN map is applied at the ingress side and at the egress side. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the ingress side. Similarly, when the frame is routed from an external port to a Private VLAN, the private-VLAN is applied at the egress side.

Bridging

- For upstream traffic from secondary VLAN to primary VLAN, the MAP of the secondary VLAN is applied on the ingress side and the MAP of the primary VLAN is applied on the egress side.
- For downstream traffic from primary VLAN to secondary VLAN, the MAP of the primary VLAN is applied in the ingress direction and the MAP of the secondary VLAN is applied in the egress direction.

Routing

If we have two private VLAN domains - PV1 (sec1, prim1) and PV2 (sec2, prim2). For frames routed from PV1 to PV2:

- The MAP of sec1 and L3 ACL of prim1 is applied in the ingress port .
- The MAP of sec1 and L3 ACL of prim2 is applied in the egress port.
- For packets going upstream or downstream from isolated host port to promiscuous port, the isolated VLAN's VACL is applied in the ingress direction and primary VLAN'S VACL is applied in the egress direction. This allows user to configure different VACL for different secondary VLAN in a same primary VLAN domain.

To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs.

- You can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.
- Although private VLANs provide host isolation at Layer 2, hosts can communicate with each other at Layer 3.
- Private VLANs support these Switched Port Analyzer (SPAN) features:
 - You can configure a private-VLAN port as a SPAN source port.
 - You can use VLAN-based SPAN (VSPAN) on primary, isolated, and community VLANs or use SPAN on only one VLAN to separately monitor egress or ingress traffic.

Private VLAN Port Configuration

Follow these guidelines when configuring private VLAN ports:

- Use only the private VLAN configuration commands to assign ports to primary, isolated, or community VLANs. Layer 2 access ports assigned to the VLANs that you configure as primary, isolated, or community VLANs are inactive while the VLAN is part of the private VLAN configuration. Layer 2 trunk interfaces remain in the STP forwarding state.
- Do not configure ports that belong to a PAGP or LACP EtherChannel as private VLAN ports. While a port is part of the private VLAN configuration, any EtherChannel configuration for it is inactive.
- Enable Port Fast and BPDU guard on isolated and community host ports to prevent STP loops due to misconfigurations and to speed up STP convergence. When enabled, STP applies the BPDU guard feature to all Port Fast-configured Layer 2 LAN ports. Do not enable Port Fast and BPDU guard on promiscuous ports.
- If you delete a VLAN used in the private VLAN configuration, the private VLAN ports associated with the VLAN become inactive.
- Private VLAN ports can be on different network devices if the devices are trunk-connected and the primary and secondary VLANs have not been removed from the trunk.

Private VLAN Configuration Tasks

To configure a private VLAN, perform these steps:

1. Set VTP mode to transparent.
2. Create the primary and secondary VLANs and associate them.



Note If the VLAN is not created already, the private VLAN configuration process creates it.

3. Configure interfaces to be isolated or community host ports, and assign VLAN membership to the host port.
4. Configure interfaces as promiscuous ports, and map the promiscuous ports to the primary-secondary VLAN pair.
5. If inter-VLAN routing will be used, configure the primary SVI, and map the secondary VLANs to the primary.
6. Verify the private VLAN configuration.

How to Configure Private VLANs

Configuring and Associating VLANs in a Private VLAN

The **private-vlan** commands do not take effect until you exit VLAN configuration mode.

To configure and associate VLANs in a Private VLAN, perform these steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp mode transparent**
4. **vlan *vlan-id***
5. **private-vlan primary**
6. **exit**
7. **vlan *vlan-id***
8. **private-vlan isolated**
9. **exit**
10. **vlan *vlan-id***
11. **private-vlan community**
12. **exit**
13. **vlan *vlan-id***
14. **private-vlan community**
15. **exit**
16. **vlan *vlan-id***
17. **private-vlan association [add | remove] *secondary_vlan_list***
18. **end**
19. **show vlan private-vlan [type] or show interfaces status**
20. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	vtp mode transparent Example: Switch(config)# vtp mode transparent	Sets VTP mode to transparent (disable VTP). Note For VTP3, you can set mode to either server or transparent mode

	Command or Action	Purpose
Step 4	vlan <i>vlan-id</i> Example: Switch(config)# vlan 20	Enters VLAN configuration mode and designates or creates a VLAN that will be the primary VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 5	private-vlan primary Example: Switch(config-vlan)# private-vlan primary	Designates the VLAN as the primary VLAN.
Step 6	exit Example: Switch(config-vlan)# exit	Returns to global configuration mode.
Step 7	vlan <i>vlan-id</i> Example: Switch(config)# vlan 501	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be an isolated VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 8	private-vlan isolated Example: Switch(config-vlan)# private-vlan isolated	Designates the VLAN as an isolated VLAN.
Step 9	exit Example: Switch(config-vlan)# exit	Returns to global configuration mode.
Step 10	vlan <i>vlan-id</i> Example: Switch(config)# vlan 502	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 11	private-vlan community Example: Switch(config-vlan)# private-vlan community	Designates the VLAN as a community VLAN.

	Command or Action	Purpose
Step 12	exit Example: <pre>Switch(config-vlan)# exit</pre>	Returns to global configuration mode.
Step 13	vlan <i>vlan-id</i> Example: <pre>Switch(config)# vlan 503</pre>	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 14	private-vlan community Example: <pre>Switch(config-vlan)# private-vlan community</pre>	Designates the VLAN as a community VLAN.
Step 15	exit Example: <pre>Switch(config-vlan)# exit</pre>	Returns to global configuration mode.
Step 16	vlan <i>vlan-id</i> Example: <pre>Switch(config)# vlan 20</pre>	Enters VLAN configuration mode for the primary VLAN designated in Step 4.
Step 17	private-vlan association [add remove] <i>secondary_vlan_list</i> Example: <pre>Switch(config-vlan)# private-vlan association 501-503</pre>	<p>Associates the secondary VLANs with the primary VLAN. It can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs.</p> <ul style="list-style-type: none"> • The <i>secondary_vlan_list</i> parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The <i>secondary_vlan_list</i> parameter can contain multiple community VLAN IDs but only one isolated VLAN ID. • Enter a <i>secondary_vlan_list</i>, or use the add keyword with a <i>secondary_vlan_list</i> to associate secondary VLANs with a primary VLAN. • Use the remove keyword with a <i>secondary_vlan_list</i> to clear the association between secondary VLANs and a primary VLAN.

	Command or Action	Purpose
		<ul style="list-style-type: none"> The command does not take effect until you exit VLAN configuration mode.
Step 18	end Example: <pre>Switch(config)# end</pre>	Returns to privileged EXEC mode.
Step 19	show vlan private-vlan [type] or show interfaces status Example: <pre>Switch# show vlan private-vlan</pre>	Verifies the configuration.
Step 20	copy running-config startup config Example: <pre>Switch# copy running-config startup-config</pre>	Saves your entries in the device startup configuration file.

Configuring a Layer 2 Interface as a Private VLAN Host Port

Follow these steps to configure a Layer 2 interface as a private-VLAN host port and to associate it with primary and secondary VLANs:



Note Isolated and community VLANs are both secondary VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan host**
5. **switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
	Switch> enable	
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/22	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 4	switchport mode private-vlan host Example: Switch(config-if)# switchport mode private-vlan host	Configures the Layer 2 port as a private-VLAN host port.
Step 5	switchport private-vlan host-association <i>primary_vlan_id secondary_vlan_id</i> Example: Switch(config-if)# switchport private-vlan host-association 20 501	Associates the Layer 2 port with a private VLAN. Note This is a required step to associate the PVLAN to a Layer 2 interface.
Step 6	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 7	show interfaces [<i>interface-id</i>] switchport Example: Switch# show interfaces gigabitethernet1/0/22 switchport	Verifies the configuration.
Step 8	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port

Follow these steps to configure a Layer 2 interface as a private VLAN promiscuous port and map it to primary and secondary VLANs:



Note Isolated and community VLANs are both secondary VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan promiscuous**
5. **switchport private-vlan mapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Switch(config)# interface gigabitethernet1/0/2</pre>	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 4	switchport mode private-vlan promiscuous Example: <pre>Switch(config-if)# switchport mode private-vlan promiscuous</pre>	Configures the Layer 2 port as a private VLAN promiscuous port.

	Command or Action	Purpose
Step 5	<p>switchport private-vlan mapping <i>primary_vlan_id</i> {add remove} <i>secondary_vlan_list</i></p> <p>Example:</p> <pre>Switch(config-if) # switchport private-vlan mapping 20 add 501-503</pre>	<p>Maps the private VLAN promiscuous port to a primary VLAN and to selected secondary VLANs.</p> <ul style="list-style-type: none"> • The <i>secondary_vlan_list</i> parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private VLAN ID or a hyphenated range of private VLAN IDs. • Enter a <i>secondary_vlan_list</i>, or use the add keyword with a <i>secondary_vlan_list</i> to map the secondary VLANs to the private VLAN promiscuous port. • Use the remove keyword with a <i>secondary_vlan_list</i> to clear the mapping between secondary VLANs and the private VLAN promiscuous port.
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config) # end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show interfaces [<i>interface-id</i>] switchport</p> <p>Example:</p> <pre>Switch# show interfaces gigabitethernet1/0/2 switchport</pre>	Verifies the configuration.
Step 8	<p>copy running-config startup config</p> <p>Example:</p> <pre>Switch# copy running-config startup-config</pre>	Saves your entries in the device startup configuration file.

Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface

If the private VLAN will be used for inter-VLAN routing, you configure an SVI for the primary VLAN and map secondary VLANs to the SVI.



Note Isolated and community VLANs are both secondary VLANs.

Follow these steps to map secondary VLANs to the SVI of a primary VLAN to allow Layer 3 switching of private VLAN traffic:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface vlan primary_vlan_id`
4. `private-vlan mapping [add | remove] secondary_vlan_list`
5. `end`
6. `show interface private-vlan mapping`
7. `copy running-config startup config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Switch> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Switch# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>interface vlan <i>primary_vlan_id</i></code></p> <p>Example:</p> <pre>Switch(config)# interface vlan 20</pre>	<p>Enters interface configuration mode for the primary VLAN, and configures the VLAN as an SVI. The VLAN ID range is 2 to 1001 and 1006 to 4094.</p>
Step 4	<p><code>private-vlan mapping [add remove] <i>secondary_vlan_list</i></code></p> <p>Example:</p> <pre>Switch(config-if)# private-vlan mapping 501-503</pre>	<p>Maps the secondary VLANs to the Layer 3 VLAN interface of a primary VLAN to allow Layer 3 switching of private VLAN ingress traffic.</p> <p>Note The private-vlan mapping interface configuration command only affects private VLAN traffic that is Layer 3 switched.</p> <ul style="list-style-type: none"> • The <i>secondary_vlan_list</i> parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • Enter a <i>secondary_vlan_list</i>, or use the add keyword with a <i>secondary_vlan_list</i> to map the secondary VLANs to a primary VLAN. • Use the remove keyword with a <i>secondary_vlan_list</i> to clear the mapping between secondary VLANs and a primary VLAN.

	Command or Action	Purpose
Step 5	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 6	show interface private-vlan mapping Example: Switch# show interfaces private-vlan mapping	Verifies the configuration.
Step 7	copy running-config startup config Example: Switch# copy running-config startup-config	Saves your entries in the device startup configuration file.

Monitoring Private VLANs

The following table displays the commands used to monitor private VLANs.

Table 214: Private VLAN Monitoring Commands

Command	Purpose
show interfaces status	Displays the status of interfaces, including
show vlan private-vlan [type]	Displays the private VLAN information fo
show interface switchport	Displays private VLAN configuration on i
show interface private-vlan mapping	Displays information about the private VL

Configuration Examples for Private VLANs

Example: Configuring an Interface as a Host Port

This example shows how to configure an interface as a private VLAN host port, associate it with a private VLAN pair, and verify the configuration:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/22
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# switchport private-vlan host-association 20 501
Switch(config-if)# end
```

```
Switch# show interfaces gigabitethernet1/0/22 switchport
Name: Gi1/0/22
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: 20 501
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan:
20 501

<output truncated>
```

Example: Configuring an Interface as a Private VLAN Promiscuous Port

This example shows how to configure an interface as a private VLAN promiscuous port and map it to a private VLAN. The interface is a member of primary VLAN 20 and secondary VLANs 501 to 503 are mapped to it.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport mode private-vlan promiscuous
Switch(config-if)# switchport private-vlan mapping 20 add 501-503
Switch(config-if)# end
```

Use the **show vlan private-vlan** or the **show interface status** privileged EXEC command to display primary and secondary VLANs and private-VLAN ports on the Switch.

Example: Mapping Secondary VLANs to a Primary VLAN Interface

This example shows how to map the interfaces for VLANs 501 and 502 to primary VLAN 10, which permits routing of secondary VLAN ingress traffic from private VLANs 501 and 502:

```
Switch# configure terminal
Switch(config)# interface vlan 20
Switch(config-if)# private-vlan mapping 501-503
Switch(config-if)# end
Switch# show interfaces private-vlan mapping
Interface Secondary VLAN Type
-----
vlan20    501        isolated
vlan20    502        community
vlan20    503        community
```

Example: Monitoring Private VLANs

This example shows output from the `show vlan private-vlan` command:

```
Switch# show vlan private-vlan
Primary Secondary Type Ports
-----
20      501      isolated      Gi1/0/22, Gi1/0/2
20      502      community     Gi1/0/2
20      503      community     Gi1/0/2
```

Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN trunking
- VLAN Membership Policy Server (VMPS)
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
CLI commands	LAN Switching Command Reference, Cisco IOS Release

Standards and RFCs

Standard/RFC	Title
RFC 1573	
RFC 1757	
RFC 2021	

MIBs

MIB	MIBs Link
<p>All the supported MIBs for this release.</p> <ul style="list-style-type: none"> • BRIDGE-MIB (RFC1493) • CISCO-BRIDGE-EXT-MIB • CISCO-CDP-MIB • CISCO-PAGP-MIB • CISCO-PRIVATE-VLAN-MIB • CISCO-LAG-MIB • CISCO-L2L3-INTERFACE-CONFIG-MIB • CISCO-MAC-NOTIFICATION-MIB • CISCO-STP-EXTENSIONS-MIB • CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB • CISCO-VLAN-MEMBERSHIP-MIB • CISCO-VTP-MIB • IEEE8023-LAG-MIB • IF-MIB (RFC 1573) • RMON-MIB (RFC 1757) • RMON2-MIB (RFC 2021) 	<p>To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>

