



Cisco Nexus 5500 Series NX-OS Security Configuration Guide, Release 6.0(2)N1(2)

First Published: 2013-03-11

Last Modified: 2013-03-11

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

Text Part Number: OL-27902-02

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.

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: <http://www.cisco.com/go/trademarks>. 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. (1110R)

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

© 2013 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface	xv
Audience	xv
Document Conventions	xv
Related Documentation for Cisco Nexus 5500 Series NX-OS Software	xvi
Documentation Feedback	xviii
Obtaining Documentation and Submitting a Service Request	xviii

CHAPTER 1

New and Changed Information	1
New and Changed Information for this Release	1

CHAPTER 2

Overview	3
Authentication, Authorization, and Accounting	3
RADIUS and TACACS+ Security Protocols	4
SSH and Telnet	4
IP ACLs	5

CHAPTER 3

Configuring Authentication, Authorization, and Accounting	7
Information About AAA	7
AAA Security Services	7
Benefits of Using AAA	8
Remote AAA Services	8
AAA Server Groups	8
AAA Service Configuration Options	8
Authentication and Authorization Process for User Logins	9
Prerequisites for Remote AAA	11
Configuring AAA	12

Configuring Console Login Authentication Methods	12
Configuring Default Login Authentication Methods	13
Enabling Login Authentication Failure Messages	14
Configuring AAA Command Authorization	14
Enabling MSCHAP Authentication	16
Configuring AAA Accounting Default Methods	17
Using AAA Server VSAs	18
VSAs	18
VSA Format	18
Specifying Switch User Roles and SNMPv3 Parameters on AAA Servers	19
Monitoring and Clearing the Local AAA Accounting Log	19
Verifying the AAA Configuration	20
Configuration Examples for AAA	20
Default AAA Settings	20
<hr/>	
CHAPTER 4	Configuring RADIUS 21
Configuring RADIUS	21
Information About RADIUS	21
RADIUS Network Environments	21
Information About RADIUS Operations	22
RADIUS Server Monitoring	22
Vendor-Specific Attributes	23
Prerequisites for RADIUS	24
Guidelines and Limitations for RADIUS	24
Configuring RADIUS Servers	24
Configuring RADIUS Server Hosts	24
Configuring RADIUS Global Preshared Keys	25
Configuring RADIUS Server Preshared Keys	26
Configuring RADIUS Server Groups	27
Configuring the Global Source Interface for RADIUS Server Groups	28
Allowing Users to Specify a RADIUS Server at Login	29
Configuring the Global RADIUS Transmission Retry Count and Timeout Interval	30
Configuring the RADIUS Transmission Retry Count and Timeout Interval for a Server	30
Configuring Accounting and Authentication Attributes for RADIUS Servers	31

Configuring Periodic RADIUS Server Monitoring	32
Configuring the Dead-Time Interval	34
Manually Monitoring RADIUS Servers or Groups	34
Verifying the RADIUS Configuration	35
Displaying RADIUS Server Statistics	35
Clearing RADIUS Server Statistics	35
Configuration Examples for RADIUS	36
Default Settings for RADIUS	36

CHAPTER 5**Configuring TACACS+ 37**

About Configuring TACACS+	37
Information About TACACS+	37
TACACS+ Advantages	37
User Login with TACACS+	38
Default TACACS+ Server Encryption Type and Preshared Key	38
Command Authorization Support for TACACS+ Servers	39
TACACS+ Server Monitoring	39
Prerequisites for TACACS+	39
Guidelines and Limitations for TACACS+	40
Configuring TACACS+	40
TACACS+ Server Configuration Process	40
Displaying TACACS+ Statistics	56
Verifying the TACACS+ Configuration	56
Configuration Examples for TACACS+	57
Default Settings for TACACS+	57

CHAPTER 6**Configuring SSH and Telnet 59**

Configuring SSH and Telnet	59
Information About SSH and Telnet	59
SSH Server	59
SSH Client	59
SSH Server Keys	59
Telnet Server	60
Guidelines and Limitations for SSH	60

Configuring SSH	60
Generating SSH Server Keys	60
Specifying the SSH Public Keys for User Accounts	61
Starting SSH Sessions to Remote Devices	63
Clearing SSH Hosts	63
Disabling the SSH Server	64
Deleting SSH Server Keys	64
Clearing SSH Sessions	65
Configuration Examples for SSH	65
Configuring Telnet	66
Enabling the Telnet Server	66
Starting Telnet Sessions to Remote Devices	66
Clearing Telnet Sessions	67
Verifying the SSH and Telnet Configuration	67
Default Settings for SSH	68

CHAPTER 7

Configuring 802.1X	69
Information About 802.1X	69
Device Roles	69
Authentication Initiation and Message Exchange	71
Authenticator PAE Status for Interfaces	72
Ports in Authorized and Unauthorized States	72
MAC Authentication Bypass	73
802.1X and Port Security	74
Single Host and Multiple Hosts Support	74
Supported Topologies	75
Licensing Requirements for 802.1X	75
Prerequisites for 802.1X	76
802.1X Guidelines and Limitations	76
Default Settings for 802.1X	77
Configuring 802.1X	78
Process for Configuring 802.1X	78
Enabling the 802.1X Feature	78
Configuring AAA Authentication Methods for 802.1X	79

Controlling 802.1X Authentication on an Interface	80
Configuring 802.1X Authentication on Member Ports	81
Creating or Removing an Authenticator PAE on an Interface	83
Enabling Periodic Reauthentication for an Interface	84
Manually Reauthenticating Supplicants	85
Manually Initializing 802.1X Authentication	86
Changing 802.1X Authentication Timers for an Interface	86
Enabling Single Host or Multiple Hosts Mode	88
Enabling MAC Authentication Bypass	89
Disabling 802.1X Authentication on the Cisco NX-OS Device	90
Disabling the 802.1X Feature	91
Setting the Maximum Authenticator-to-Supplicant Frame Retransmission Retry Count for an Interface	92
Enabling RADIUS Accounting for 802.1X Authentication	93
Configuring AAA Accounting Methods for 802.1X	94
Setting the Maximum Reauthentication Retry Count on an Interface	95
Verifying the 802.1X Configuration	96
Monitoring 802.1X	96
Configuration Example for 802.1X	96
Additional References for 802.1X	97
Feature History for 802.1X	98

CHAPTER 8**Configuring Cisco TrustSec 99**

Information About Cisco TrustSec	99
Cisco TrustSec Architecture	99
Authentication	100
Device Identities	100
Device Credentials	101
User Credentials	101
SGACLs and SGTs	101
Determining the Source Security Group	102
Determining the Destination Security Group	103
SXP for SGT Propagation Across Legacy Access Networks	103
Environment Data Download	104

Licensing Requirements for Cisco TrustSec	104
Prerequisites for Cisco TrustSec	105
Guidelines and Limitations for Cisco TrustSec	105
Default Settings For Cisco TrustSec	106
Configuring Cisco TrustSec	106
Enabling the Cisco TrustSec Feature	106
Configuring Cisco TrustSec Device Credentials	107
Configuring AAA for Cisco TrustSec	108
Configuring AAA on the Cisco TrustSec Cisco NX-OS Devices	108
Configuring Cisco TrustSec Authentication in Manual Mode	111
Configuring Pause Frame Encryption or Decryption for Cisco TrustSec on Interfaces	113
Configuring SGACL Policies	114
SGACL Policy Configuration Process	114
Enabling SGACL Policy Enforcement on VLANs	115
Manually Configuring Cisco TrustSec SGTs	116
Manually Configuring IPv4-Address-to-SGACL SGT Mapping for a VLAN	117
Manually Configuring IPv4-Address-to-SGACL SGT Mapping for a VRF	118
Manually Configuring SGACL Policies	119
Displaying the Downloaded SGACL Policies	121
Refreshing the Downloaded SGACL Policies	121
Enabling Statistics for RBACL	121
Clearing Cisco TrustSec SGACL Policies	122
Manually Configuring SXP	123
Cisco TrustSec SXP Configuration Process	123
Enabling Cisco TrustSec SXP	124
Configuring Cisco TrustSec SXP Peer Connections	124
Configuring the Default SXP Password	126
Configuring the Default SXP Source IPv4 Address	127
Changing the SXP Retry Period	128
Verifying the Cisco TrustSec Configuration	129
Configuration Examples for Cisco TrustSec	130
Enabling Cisco TrustSec	130
Configuring AAA for Cisco TrustSec on a Cisco NX-OS Device	130
Configuring Cisco TrustSec Authentication in Manual Mode	130

Configuring Cisco TrustSec Role-Based Policy Enforcement for a VLAN	130
Configuring IPv4 Address to SGACL SGT Mapping for the Default VRF	131
Configuring IPv4 Address to SGACL SGT Mapping for a VLAN	131
Manually Configuring Cisco TrustSec SGACLs	131
Manually Configuring SXP Peer Connections	132
Additional References for Cisco TrustSec	133
Feature History for Cisco TrustSec	133

CHAPTER 9
Configuring Access Control Lists 135

Information About ACLs	135
IP ACL Types and Applications	135
Application Order	136
Rules	137
Source and Destination	137
Protocols	137
Implicit Rules	137
Additional Filtering Options	137
Sequence Numbers	138
Logical Operators and Logical Operation Units	139
Statistics and ACLs	139
Licensing Requirements for ACLs	140
Prerequisites for ACLs	140
Guidelines and Limitations for ACLs	140
Default ACL Settings	141
Configuring IP ACLs	142
Creating an IP ACL	142
Changing an IP ACL	143
Removing an IP ACL	144
Changing Sequence Numbers in an IP ACL	144
Configuring ACLs with Logging	145
Applying an IP ACL to mgmt0	146
Applying an IP ACL as a Router ACL	146
Applying an IP ACL as a Port ACL	147
Verifying IP ACL Configurations	148

Monitoring and Clearing IP ACL Statistics	148
Configuring MAC ACLs	149
Creating a MAC ACL	149
Changing a MAC ACL	150
Removing a MAC ACL	151
Changing Sequence Numbers in a MAC ACL	151
Applying a MAC ACL as a Port ACL	152
Verifying MAC ACL Configurations	153
Displaying and Clearing MAC ACL Statistics	153
Example Configuration for MAC ACLs	153
Information About VLAN ACLs	154
VACLs and Access Maps	154
VACLs and Actions	154
Statistics	154
Configuring VACLs	154
Creating or Changing a VACL	154
Removing a VACL	155
Applying a VACL to a VLAN	155
Verifying VACL Configuration	156
Displaying and Clearing VACL Statistics	156
Configuration Examples for VACL	157
Configuring ACLs on Virtual Terminal Lines	157
Verifying ACLs on VTY Lines	158
Configuration Examples for ACLs on VTY Lines	159

CHAPTER 10**Configuring Port Security 161**

Information About Port Security	161
Secure MAC Address Learning	161
Static Method	162
Dynamic Method	162
Sticky Method	162
Dynamic Address Aging	163
Secure MAC Address Maximums	163
Security Violations and Actions	164

Port Type Changes	165
Licensing Requirements for Port Security	166
Prerequisites for Port Security	166
Guidelines and Limitations for Port Security	166
Guidelines and Limitations for Port Security on vPCs	166
Configuring Port Security	167
Enabling or Disabling Port Security Globally	167
Enabling or Disabling Port Security on a Layer 2 Interface	168
Enabling or Disabling Sticky MAC Address Learning	169
Adding a Static Secure MAC Address on an Interface	170
Removing a Static Secure MAC Address on an Interface	171
Removing a Dynamic Secure MAC Address	172
Configuring a Maximum Number of MAC Addresses	173
Configuring an Address Aging Type and Time	174
Configuring a Security Violation Action	175
Verifying the Port Security Configuration	176
Displaying Secure MAC Addresses	177
Configuration Example for Port Security	177
Configuration Example of Port Security in a vPC Domain	177
Default Settings for Port Security	178
Additional References for Port Security	178
Feature History for Port Security	179

CHAPTER 11
Configuring DHCP Snooping 181

Information About DHCP Snooping	181
Feature Enabled and Globally Enabled	181
Trusted and Untrusted Sources	182
DHCP Snooping Binding Database	183
DHCP Snooping Option 82 Data Insertion	183
DHCP Snooping in a vPC Environment	185
Synchronizing DHCP Snooping Binding Entries	185
Packet Validation	185
Information About the DHCP Relay Agent	186
DHCP Relay Agent	186

VRF Support for the DHCP Relay Agent	186
DHCP Relay Binding Database	187
Guidelines and Limitations for DHCP Snooping	187
Default Settings for DHCP Snooping	188
Configuring DHCP Snooping	188
Minimum DHCP Snooping Configuration	188
Enabling or Disabling the DHCP Snooping Feature	189
Enabling or Disabling DHCP Snooping Globally	190
Enabling or Disabling DHCP Snooping on a VLAN	190
Enabling or Disabling Option 82 Data Insertion and Removal	191
Enabling or Disabling Strict DHCP Packet Validation	192
Configuring an Interface as Trusted or Untrusted	192
Enabling or Disabling the DHCP Relay Agent	193
Enabling or Disabling Option 82 for the DHCP Relay Agent	194
Enabling or Disabling VRF Support for the DHCP Relay Agent	195
Enabling or Disabling Subnet Broadcast Support for the DHCP Relay Agent on a Layer 3 Interface	196
Creating a DHCP Static Binding	197
Verifying the DHCP Snooping Configuration	198
Displaying DHCP Bindings	198
Clearing the DHCP Snooping Binding Database	199
Configuration Examples for DHCP Snooping	200
<hr/>	
CHAPTER 12	Configuring Dynamic ARP Inspection 201
Information About DAI	201
ARP	201
ARP Spoofing Attacks	201
DAI and ARP Spoofing Attacks	202
Interface Trust States and Network Security	203
Logging DAI Packets	204
Licensing Requirements for DAI	204
Prerequisites for DAI	205
Guidelines and Limitations for DAI	205
Default Settings for DAI	206

Configuring DAI	206
Enabling or Disabling DAI on VLANs	206
Configuring the DAI Trust State of a Layer 2 Interface	207
Enabling or Disabling Additional Validation	208
Configuring the DAI Logging Buffer Size	209
Configuring DAI Log Filtering	210
Verifying the DAI Configuration	211
Monitoring and Clearing DAI Statistics	211
Configuration Examples for DAI	212
Example 1-Two Devices Support DAI	212
Configuring Device A	212
Configuring Device B	214
<hr/>	
CHAPTER 13	Configuring IP Source Guard 217
	Information About IP Source Guard 217
	Licensing Requirements for IP Source Guard 218
	Prerequisites for IP Source Guard 218
	Guidelines and Limitations for IP Source Guard 218
	Default Settings for IP Source Guard 219
	Configuring IP Source Guard 219
	Enabling or Disabling IP Source Guard on a Layer 2 Interface 219
	Adding or Removing a Static IP Source Entry 220
	Displaying IP Source Guard Bindings 221
	Configuration Example for IP Source Guard 221
	Additional References for IP Source Guard 221
<hr/>	
CHAPTER 14	Configuring Control Plane Policing 223
	Information About CoPP 223
	Control Plane Protection 224
	Control Plane Packet Types 224
	Classification for CoPP 225
	Rate Controlling Mechanisms 225
	CoPP Class Maps 225
	CoPP Policy Templates 228

Default CoPP Policy	229
Scaled Layer 2 CoPP Policy	230
Scaled Layer 3 CoPP Policy	231
Customizable CoPP Policy	232
CoPP and the Management Interface	233
Licensing Requirements for CoPP	233
Guidelines and Limitations for CoPP	233
Default Settings for CoPP	234
Configuring CoPP	234
Applying a CoPP Policy to the Switch	234
Modifying the Customized CoPP Policy	235
Verifying the CoPP Configuration	236
Displaying the CoPP Configuration Status	236
Monitoring CoPP	237
Clearing the CoPP Statistics	237
Additional References for CoPP	238
Feature History for CoPP	238



Preface

This preface contains the following sections:

- [Audience, on page xv](#)
- [Document Conventions, on page xv](#)
- [Related Documentation for Cisco Nexus 5500 Series NX-OS Software, on page xvi](#)
- [Documentation Feedback , on page xviii](#)
- [Obtaining Documentation and Submitting a Service Request, on page xviii](#)

Audience

This publication is for network administrators who configure and maintain Cisco Nexus devices and Cisco Nexus 2000 Series Fabric Extenders.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
<i>Italic</i>	Italic text indicates arguments for which the user supplies the values.
[x]	Square brackets enclose an optional element(keyword or argument).
[x y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.
[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.

Convention	Description
variable	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.

Examples use the following conventions:

Convention	Description
screen font	Terminal sessions and information the switch displays are in screen font.
boldface screen font	Information you must enter is in boldface screen font.
<i>italic screen font</i>	Arguments for which you supply values are in italic screen font.
<>	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.

This document uses the following conventions:



Note

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



Caution

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

Related Documentation for Cisco Nexus 5500 Series NX-OS Software

The entire Cisco NX-OS 5500 Series documentation set is available at the following URL:

http://www.cisco.com/en/US/products/ps9670/tsd_products_support_series_home.html

Release Notes

The release notes are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/prod_release_notes_list.html

Configuration Guides

These guides are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/products_installation_and_configuration_guides_list.html

The documents in this category include:

- *Cisco Nexus 5500 Series NX-OS Adapter-FEX Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS FabricPath Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS FCoE Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Fundamentals Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Interfaces Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Layer 2 Switching Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Multicast Routing Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Quality of Service Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS SAN Switching Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Security Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS System Management Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Unicast Routing Configuration Guide*

Installation and Upgrade Guides

These guides are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/prod_installation_guides_list.html

The document in this category include:

- *Cisco Nexus 5500 Series NX-OS Software Upgrade and Downgrade Guides*

Licensing Guide

The *License and Copyright Information for Cisco NX-OS Software* is available at

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/license_agreement/nx-ossw_lisns.html.

Command References

These guides are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/prod_command_reference_list.html

The documents in this category include:

- *Cisco Nexus 5500 Series NX-OS Fabric Extender Command Reference*
- *Cisco Nexus 5500 Series NX-OS FabricPath Command Reference*
- *Cisco Nexus 5500 Series NX-OS Fundamentals Command Reference*
- *Cisco Nexus 5500 Series NX-OS Interfaces Command Reference*
- *Cisco Nexus 5500 Series NX-OS Layer 2 Interfaces Command Reference*

- *Cisco Nexus 5500 Series NX-OS Multicast Routing Command Reference*
- *Cisco Nexus 5500 Series NX-OS Quality of Service Command Reference*
- *Cisco Nexus 5500 Series NX-OS Security Command Reference*
- *Cisco Nexus 5500 Series NX-OS System Management Command Reference*
- *Cisco Nexus 5500 Series NX-OS TrustSec Command Reference*
- *Cisco Nexus 5500 Series NX-OS Unicast Routing Command Reference*
- *Cisco Nexus 5500 Series NX-OS Virtual Port Channel Command Reference*

Technical References

The *Cisco Nexus 5500 Series NX-OS MIB Reference* is available at http://www.cisco.com/en/US/docs/switches/datacenter/nexus5500/sw/mib/reference/NX5500_MIBRef.html.

Error and System Messages

The *Cisco Nexus 5500 Series NX-OS System Message Guide* is available at http://www.cisco.com/en/US/docs/switches/datacenter/nexus5500/sw/system_messages/reference/sl_nxos_book.html.

Troubleshooting Guide

The *Cisco Nexus 5500 Series NX-OS Troubleshooting Guide* is available at http://www.cisco.com/en/US/docs/switches/datacenter/nexus5500/sw/troubleshooting/guide/N5K_Troubleshooting_Guide.html.

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus5k-docfeedback@cisco.com. We appreciate your feedback.

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/en/US/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.



CHAPTER 1

New and Changed Information

This chapter contains the following sections:

- [New and Changed Information for this Release, on page 1](#)

New and Changed Information for this Release

The following table provides an overview of the significant changes to this guide for this current release. The table does not provide an exhaustive list of all changes made to the configuration guides or of the new features in this release.

Table 1: New Features

Feature	Description	Where Documented
802.1X	802.1X defines a client-server access control and authentication protocol that restricts unauthorized clients from connecting to a LAN through publicly accessible ports.	Configuring 802.1X



CHAPTER 2

Overview

The Cisco NX-OS software supports security features that can protect your network against degradation or failure and also against data loss or compromise resulting from intentional attacks and from unintended but damaging mistakes by well-meaning network users.

- [Authentication, Authorization, and Accounting, on page 3](#)
- [RADIUS and TACACS+ Security Protocols, on page 4](#)
- [SSH and Telnet, on page 4](#)
- [IP ACLs, on page 5](#)

Authentication, Authorization, and Accounting

Authentication, authorization, and accounting (AAA) is an architectural framework for configuring a set of three independent security functions in a consistent, modular manner.

Authentication

Provides the method of identifying users, including login and password dialog, challenge and response, messaging support, and, depending on the security protocol that you select, encryption. Authentication is the way a user is identified prior to being allowed access to the network and network services. You configure AAA authentication by defining a named list of authentication methods and then applying that list to various interfaces.

Authorization

Provides the method for remote access control, including one-time authorization or authorization for each service, per-user account list and profile, user group support, and support of IP, IPX, ARA, and Telnet.

Remote security servers, such as RADIUS and TACACS+, authorize users for specific rights by associating attribute-value (AV) pairs, which define those rights, with the appropriate user. AAA authorization works by assembling a set of attributes that describe what the user is authorized to perform. These attributes are compared with the information contained in a database for a given user, and the result is returned to AAA to determine the user's actual capabilities and restrictions.

Accounting

Provides the method for collecting and sending security server information used for billing, auditing, and reporting, such as user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes. Accounting enables you to track the services that users are accessing, as well as the amount of network resources that they are consuming.



Note You can configure authentication outside of AAA. However, you must configure AAA if you want to use RADIUS or TACACS+, or if you want to configure a backup authentication method.

Related Topics

[Configuring AAA](#)

RADIUS and TACACS+ Security Protocols

AAA uses security protocols to administer its security functions. If your router or access server is acting as a network access server, AAA is the means through which you establish communication between your network access server and your RADIUS or TACACS+ security server.

The chapters in this guide describe how to configure the following security server protocols:

RADIUS

A distributed client/server system implemented through AAA that secures networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco routers and send authentication requests to a central RADIUS server that contains all user authentication and network service access information.

TACACS+

A security application implemented through AAA that provides a centralized validation of users who are attempting to gain access to a router or network access server. TACACS+ services are maintained in a database on a TACACS+ daemon running, typically, on a UNIX or Windows NT workstation. TACACS+ provides for separate and modular authentication, authorization, and accounting facilities.

Related Topics

[Configuring RADIUS](#)

[Configuring TACACS+](#), on page 37

SSH and Telnet

You can use the Secure Shell (SSH) server to enable an SSH client to make a secure, encrypted connection to a Cisco NX-OS device. SSH uses strong encryption for authentication. The SSH server in the Cisco NX-OS software can interoperate with publicly and commercially available SSH clients.

The SSH client in the Cisco NX-OS software works with publicly and commercially available SSH servers.

The Telnet protocol enables TCP/IP connections to a host. Telnet allows a user at one site to establish a TCP connection to a login server at another site and then passes the keystrokes from one device to the other. Telnet can accept either an IP address or a domain name as the remote device address.

Related Topics

[Configuring SSH and Telnet](#), on page 59

IP ACLs

IP ACLs are ordered sets of rules that you can use to filter traffic based on IPv4 information in the Layer 3 header of packets. Each rule specifies a set of conditions that a packet must satisfy to match the rule. When the Cisco NX-OS software determines that an IP ACL applies to a packet, it tests the packet against the conditions of all rules. The first match determines whether a packet is permitted or denied, or if there is no match, the Cisco NX-OS software applies the applicable default rule. The Cisco NX-OS software continues processing packets that are permitted and drops packets that are denied.

Related Topics

[Configuring IP ACLs](#)



CHAPTER 3

Configuring Authentication, Authorization, and Accounting

This chapter contains the following sections:

- [Information About AAA, on page 7](#)
- [Prerequisites for Remote AAA, on page 11](#)
- [Configuring AAA, on page 12](#)
- [Monitoring and Clearing the Local AAA Accounting Log , on page 19](#)
- [Verifying the AAA Configuration, on page 20](#)
- [Configuration Examples for AAA, on page 20](#)
- [Default AAA Settings, on page 20](#)

Information About AAA

AAA Security Services

The authentication, authorization, and accounting (AAA) features allows you to verify the identity of, grant access to, and track the actions of users who manage Cisco Nexus devices. The Cisco Nexus device supports Remote Access Dial-In User Service (RADIUS) or Terminal Access Controller Access Control device Plus (TACACS+) protocols.

Based on the user ID and password that you provide, the switches perform local authentication or authorization using the local database or remote authentication or authorization using one or more AAA servers. A preshared secret key provides security for communication between the switch and AAA servers. You can configure a common secret key for all AAA servers or for only a specific AAA server.

AAA security provides the following services:

- **Authentication**—Identifies users, including login and password dialog, challenge and response, messaging support, and, encryption depending on the security protocol that you select.
- **Authorization**—Provides access control.

Authorization to access a Cisco Nexus device is provided by attributes that are downloaded from AAA servers. Remote security servers, such as RADIUS and TACACS+, authorize users for specific rights by associating attribute-value (AV) pairs, which define those rights with the appropriate user.

- Accounting—Provides the method for collecting information, logging the information locally, and sending the information to the AAA server for billing, auditing, and reporting.



Note The Cisco NX-OS software supports authentication, authorization, and accounting independently. For example, you can configure authentication and authorization without configuring accounting.

Benefits of Using AAA

AAA provides the following benefits:

- Increased flexibility and control of access configuration
- Scalability
- Standardized authentication methods, such as RADIUS and TACACS+
- Multiple backup devices

Remote AAA Services

Remote AAA services provided through RADIUS and TACACS+ protocols have the following advantages over local AAA services:

- User password lists for each switch in the fabric are easier to manage.
- AAA servers are already deployed widely across enterprises and can be easily used for AAA services.
- The accounting log for all switches in the fabric can be centrally managed.
- User attributes for each switch in the fabric are easier to manage than using the local databases on the switches.

AAA Server Groups

You can specify remote AAA servers for authentication, authorization, and accounting using server groups. A server group is a set of remote AAA servers that implement the same AAA protocol. A server group provides for failover servers if a remote AAA server fails to respond. If the first remote server in the group fails to respond, the next remote server in the group is tried until one of the servers sends a response. If all the AAA servers in the server group fail to respond, that server group option is considered a failure. If required, you can specify multiple server groups. If a switch encounters errors from the servers in the first group, it tries the servers in the next server group.

AAA Service Configuration Options

On Cisco Nexus devices, you can have separate AAA configurations for the following services:

- User Telnet or Secure Shell (SSH) login authentication
- Console login authentication

- User management session accounting

The following table lists the CLI commands for each AAA service configuration option.

Table 2: AAA Service Configuration Commands

AAA Service Configuration Option	Related Command
Telnet or SSH login	aaa authentication login default
Console login	aaa authentication login console
User session accounting	aaa accounting default

You can specify the following authentication methods for the AAA services:

- RADIUS server groups—Uses the global pool of RADIUS servers for authentication.
- Specified server groups—Uses specified RADIUS or TACACS+ server groups for authentication.
- Local—Uses the local username or password database for authentication.
- None—Uses only the username.



Note If the method is for all RADIUS servers, instead of a specific server group, the Cisco Nexus devices choose the RADIUS server from the global pool of configured RADIUS servers in the order of configuration. Servers from this global pool are the servers that can be selectively configured in a RADIUS server group on the Cisco Nexus devices.

The following table describes the AAA authentication methods that you can configure for the AAA services.

Table 3: AAA Authentication Methods for AAA Services

AAA Service	AAA Methods
Console login authentication	Server groups, local, and none
User login authentication	Server groups, local, and none
User management session accounting	Server groups and local



Note For console login authentication, user login authentication, and user management session accounting, the Cisco Nexus devices try each option in the order specified. The local option is the default method when other configured options fail.

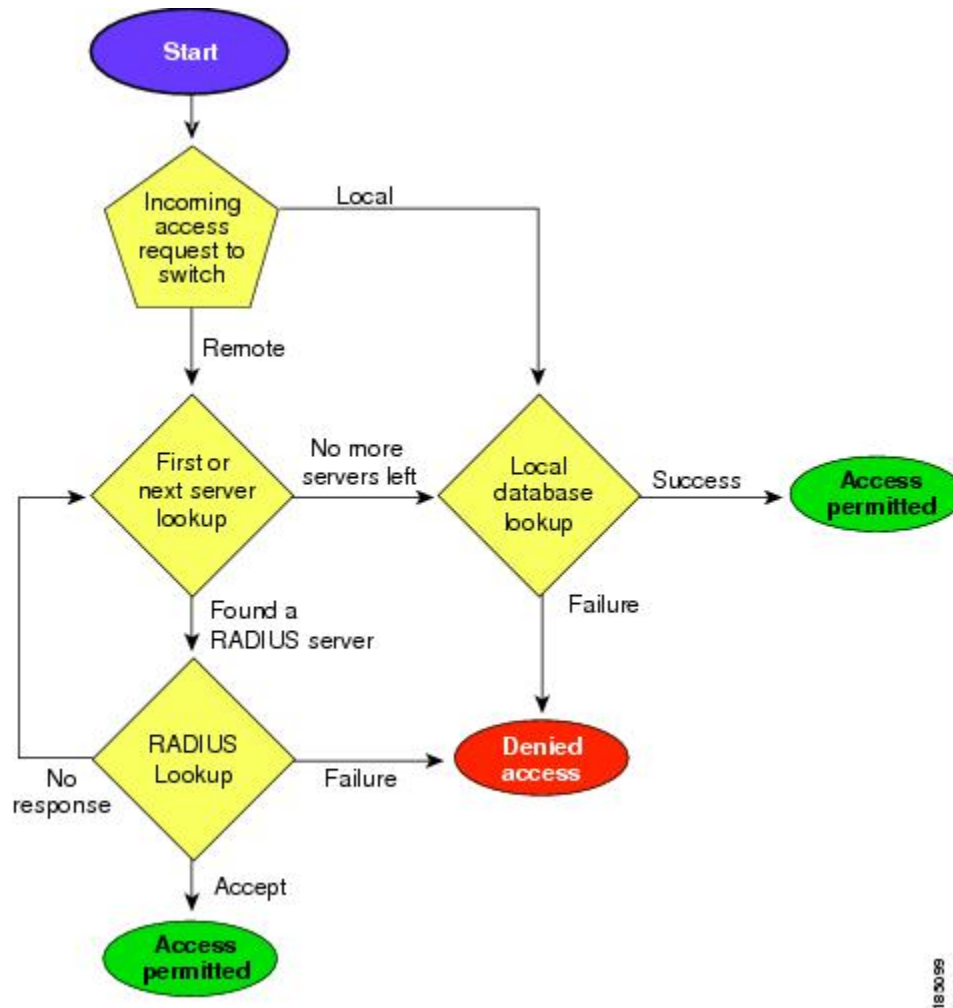
Authentication and Authorization Process for User Logins

The authentication and authorization process for user login is as occurs:

- When you log in to the required Cisco Nexus device, you can use the Telnet, SSH, Fabric Manager or Device Manager, or console login options.
- When you have configured the AAA server groups using the server group authentication method, the Cisco Nexus device sends an authentication request to the first AAA server in the group as follows:
 - If the AAA server fails to respond, then the next AAA server is tried and so on until the remote server responds to the authentication request.
 - If all AAA servers in the server group fail to respond, the servers in the next server group are tried.
 - If all configured methods fail, the local database is used for authentication.
- If a Cisco Nexus device successfully authenticates you through a remote AAA server, the following conditions apply:
 - If the AAA server protocol is RADIUS, user roles specified in the cisco-av-pair attribute are downloaded with an authentication response.
 - If the AAA server protocol is TACACS+, another request is sent to the same server to get the user roles specified as custom attributes for the shell.
- If your username and password are successfully authenticated locally, the Cisco Nexus device logs you in and assigns you the roles configured in the local database.

The following figure shows a flowchart of the authentication and authorization process.

Figure 1: Authentication and Authorization Flow for User Login



Note "No more server groups left" means that there is no response from any server in all server groups.
 "No more servers left" means that there is no response from any server within this server group.

Prerequisites for Remote AAA

Remote AAA servers have the following prerequisites:

- At least one RADIUS or TACACS+ server must be IP reachable.
- The Cisco Nexus device is configured as a client of the AAA servers.
- The preshared secret key is configured on the Cisco Nexus device and on the remote AAA servers.
- The remote server responds to AAA requests from the Cisco Nexus device.

Configuring AAA

Configuring Console Login Authentication Methods

The authentication methods include the following:

- Global pool of RADIUS servers
- Named subset of RADIUS or TACACS+ servers
- Local database on the Cisco Nexus device.
- Username only **none**

The default method is local.



Note The **group radius** and **group server-name** forms of the **aaa authentication** command are used for a set of previously defined RADIUS servers. Use the **radius server-host** command to configure the host servers. Use the **aaa group server radius** command to create a named group of servers.

Before you configure console login authentication methods, configure RADIUS or TACACS+ server groups as needed.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# aaa authentication login console {group group-list [none] local none}	<p>Configures login authentication methods for the console.</p> <p>The <i>group-list</i> argument consists of a space-delimited list of group names. The group names are the following:</p> <ul style="list-style-type: none"> • radius —Uses the global pool of RADIUS servers for authentication. • <i>named-group</i> —Uses a named subset of TACACS+ or RADIUS servers for authentication. <p>The local method uses the local database for authentication. The none method uses the username only.</p> <p>The default console login method is local, which is used when no methods are configured or when all of the configured methods fail to respond.</p>

	Command or Action	Purpose
Step 3	switch(config)# exit	Exits global configuration mode.
Step 4	(Optional) switch# show aaa authentication	Displays the configuration of the console login authentication methods.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to configure authentication methods for the console login:

```
switch# configure terminal
switch(config)# aaa authentication login console group radius
switch(config)# exit
switch# show aaa authentication
switch# copy running-config startup-config
```

Configuring Default Login Authentication Methods

The default method is local.

Before you configure default login authentication methods, configure RADIUS or TACACS+ server groups as needed.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# aaa authentication login default {group <i>group-list</i> [none] local none}	<p>Configures the default authentication methods. The <i>group-list</i> argument consists of a space-delimited list of group names. The group names are the following:</p> <ul style="list-style-type: none"> • radius —Uses the global pool of RADIUS servers for authentication. • named-group —Uses a named subset of TACACS+ or RADIUS servers for authentication. <p>The local method uses the local database for authentication. The none method uses the username only.</p> <p>The default login method is local, which is used when no methods are configured or when all of the configured methods do not respond.</p>

	Command or Action	Purpose
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show aaa authentication	Displays the configuration of the default login authentication methods.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling Login Authentication Failure Messages

When you log in, the login is processed by the local user database if the remote AAA servers do not respond. If you have enabled the displaying of login failure messages, the following message is displayed:

```
Remote AAA servers unreachable; local authentication done.
Remote AAA servers unreachable; local authentication failed.
```

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# aaa authentication login error-enable	Enables login authentication failure messages. The default is disabled.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show aaa authentication	Displays the login failure message configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring AAA Command Authorization

When a TACACS+ server authorization method is configured, you can authorize every command that a user executes with the TACACS+ server which includes all EXEC mode commands and all configuration mode commands.

The authorization methods include the following:

- Group—TACACS+ server group
- Local—Local role-based authorization
- None—No authorization is performed

The default method is Local.



Note There is no authorization on the console session.

Before you begin

You must enable TACACS+ before configuring AAA command authorization.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	aaa authorization {commands config-commands} {default} {[group group-name] [local]} {[group group-name] [none]} Example: <pre>switch(config)# aaa authorization config-commands default group tac1</pre> Example: <pre>switch# aaa authorization commands default group tac1</pre>	Configures authorization parameters. Use the commands keyword to authorize EXEC mode commands. Use the config-commands keyword to authorize configuration mode commands. Use the group , local , or none keywords to identify the authorization method.

Example

The following example shows how to authorize EXEC mode commands with TACACS+ server group *tac1*:

```
switch# aaa authorization commands default group tac1
```

The following example shows how to authorize configuration mode commands with TACACS+ server group *tac1*:

```
switch(config)# aaa authorization config-commands default group tac1
```

The following example shows how to authorize configuration mode commands with TACACS+ server group *tac1*:

- If the server is reachable, the command is allowed or not allowed based on the server response.
- If there is an error reaching the server, the command is authorized based on the user's *local* role.

```
switch(config)# aaa authorization config-commands default group tac1 local
```

The following example shows how to authorize configuration mode commands with TACACS+ server group *tac1*:

- If the server is reachable, the command is allowed or not allowed based on the server response.
- If there is an error reaching the server, allow the command regardless of the local role.

```
switch# aaa authorization commands default group tac1 none
```

The following example shows how to authorize EXEC mode commands regardless of the local role:

```
switch# aaa authorization commands default none
```

The following example shows how to authorize EXEC mode commands using the local role for authorization:

```
switch# aaa authorization commands default local
```

Enabling MSCHAP Authentication

Microsoft Challenge Handshake Authentication Protocol (MSCHAP) is the Microsoft version of CHAP. You can use MSCHAP for user logins to a Cisco Nexus device through a remote authentication server (RADIUS or TACACS+).

By default, the Cisco Nexus device uses Password Authentication Protocol (PAP) authentication between the switch and the remote server. If you enable MSCHAP, you must configure your RADIUS server to recognize the MSCHAP vendor-specific attributes (VSAs).

The following table describes the RADIUS VSAs required for MSCHAP.

Table 4: MSCHAP RADIUS VSAs

Vendor-ID Number	Vendor-Type Number	VSA	Description
311	11	MSCHAP-Challenge	Contains the challenge sent by an AAA server to an MSCHAP user. It can be used in both Access-Request and Access-Challenge packets.
211	11	MSCHAP-Response	Contains the response value provided by an MSCHAP user in response to the challenge. It is only used in Access-Request packets.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# aaa authentication login mschap enable	Enables MS-CHAP authentication. The default is disabled.
Step 3	switch(config)# exit	Exits configuration mode.

	Command or Action	Purpose
Step 4	(Optional) switch# show aaa authentication login mschap	Displays the MS-CHAP configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[VSAs](#), on page 18

Configuring AAA Accounting Default Methods

The Cisco Nexus device supports TACACS+ and RADIUS methods for accounting. The switches report user activity to TACACS+ or RADIUS security servers in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the AAA server.

When you activate AAA accounting, the Cisco Nexus device reports these attributes as accounting records, which are then stored in an accounting log on the security server.

You can create default method lists defining specific accounting methods, which include the following:

- RADIUS server group—Uses the global pool of RADIUS servers for accounting.
- Specified server group—Uses a specified RADIUS or TACACS+ server group for accounting.
- Local—Uses the local username or password database for accounting.



Note If you have configured server groups and the server groups do not respond, by default, the local database is used for authentication.

Before you begin

Before you configure AAA accounting default methods, configure RADIUS or TACACS+ server groups as needed.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# aaa accounting default {group group-list local}	Configures the default accounting method. One or more server group names can be specified in a space-separated list. The <i>group-list</i> argument consists of a space-delimited list of group names. The group names are the following: <ul style="list-style-type: none"> • radius—Uses the global pool of RADIUS servers for accounting.

	Command or Action	Purpose
		<ul style="list-style-type: none"> <i>named-group</i> —Uses a named subset of TACACS+ or RADIUS servers for accounting. <p>The local method uses the local database for accounting.</p> <p>The default method is local, which is used when no server groups are configured or when all the configured server group do not respond.</p>
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show aaa accounting	Displays the configuration AAA accounting default methods.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Using AAA Server VSAs

VSAs

You can use vendor-specific attributes (VSAs) to specify the Cisco Nexus device user roles and SNMPv3 parameters on AAA servers.

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating VSAs between the network access server and the RADIUS server. The IETF uses attribute 26. VSAs allow vendors to support their own extended attributes that are not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option using the format recommended in the specification. The Cisco vendor ID is 9, and the supported option is vendor type 1, which is named `cisco-av-pair`. The value is a string with the following format:

```
protocol : attribute seperator value *
```

The protocol is a Cisco attribute for a particular type of authorization, separator is an equal sign (=) for mandatory attributes, and an asterisk (*) indicates optional attributes.

When you use RADIUS servers for authentication on a Cisco Nexus device, the RADIUS protocol directs the RADIUS server to return user attributes, such as authorization information, with authentication results. This authorization information is specified through VSAs.

VSA Format

The following VSA protocol options are supported by the Cisco Nexus device:

- Shell— Used in access-accept packets to provide user profile information.
- Accounting—Used in accounting-request packets. If a value contains any white spaces, put it within double quotation marks.

The following attributes are supported by the Cisco Nexus device:

- **roles**—Lists all the roles assigned to the user. The value field is a string that stores the list of group names delimited by white space.
- **accountinginfo**—Stores additional accounting information in addition to the attributes covered by a standard RADIUS accounting protocol. This attribute is sent only in the VSA portion of the Account-Request frames from the RADIUS client on the switch, and it can only be used with the accounting protocol-related PDUs.

Specifying Switch User Roles and SNMPv3 Parameters on AAA Servers

You can use the VSA `cisco-av-pair` on AAA servers to specify user role mapping for the Cisco Nexus device using this format:

```
shell:roles="roleA roleB ..."
```

If you do not specify the role option in the `cisco-av-pair` attribute, the default user role is `network-operator`.



Note For information on Cisco Unified Wireless Network TACACS+ configurations and to change the user roles, see [Cisco Unified Wireless Network TACACS+ Configuration](#).

You can also specify your SNMPv3 authentication and privacy protocol attributes as follows:

```
shell:roles="roleA roleB..." snmpv3:auth=SHA priv=AES-128
```

The SNMPv3 authentication protocol options are SHA and MD5. The privacy protocol options are AES-128 and DES. If you do not specify these options in the `cisco-av-pair` attribute, MD5 and DES are the default authentication protocols.

For additional information, see the Configuring User Accounts and RBAC chapter in the System Management Configuration Guide for your Cisco Nexus device.

Monitoring and Clearing the Local AAA Accounting Log

The Cisco Nexus device maintains a local log for the AAA accounting activity.

Procedure

	Command or Action	Purpose
Step 1	switch# show accounting log [<i>size</i>] [<i>start-time</i>] <i>year month day hh : mm : ss</i>	Displays the accounting log contents. By default, the command output contains up to 250,000 bytes of the accounting log. You can use the size argument to limit command output. The range is from 0 to 250000 bytes. You can also specify a start time for the log output.
Step 2	(Optional) switch# clear accounting log	Clears the accounting log contents.

Verifying the AAA Configuration

To display AAA configuration information, perform one of the following tasks:

Command	Purpose
<code>show aaa accounting</code>	Displays AAA accounting configuration.
<code>show aaa authentication [login {error-enable mschap}]</code>	Displays AAA authentication information.
<code>show aaa authorization</code>	Displays AAA authorization information.
<code>show aaa groups</code>	Displays the AAA server group configuration.
<code>show running-config aaa [all]</code>	Displays the AAA configuration in the running configuration.
<code>show startup-config aaa</code>	Displays the AAA configuration in the startup configuration.

Configuration Examples for AAA

The following example shows how to configure AAA:

```
switch(config)# aaa authentication login default group radius
switch(config)# aaa authentication login console group radius
switch(config)# aaa accounting default group radius
```

Default AAA Settings

The following table lists the default settings for AAA parameters.

Table 5: Default AAA Parameters

Parameters	Default
Console authentication method	local
Default authentication method	local
Login authentication failure messages	Disabled
MSCHAP authentication	Disabled
Default accounting method	local
Accounting log display length	250 KB



CHAPTER 4

Configuring RADIUS

This chapter contains the following sections:

- [Configuring RADIUS, on page 21](#)

Configuring RADIUS

Information About RADIUS

The Remote Access Dial-In User Service (RADIUS) distributed client/server system allows you to secure networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco Nexus devices and send authentication and accounting requests to a central RADIUS server that contains all user authentication and network service access information.

RADIUS Network Environments

RADIUS can be implemented in a variety of network environments that require high levels of security while maintaining network access for remote users.

You can use RADIUS in the following network environments that require access security:

- Networks with multiple-vendor network devices, each supporting RADIUS.

For example, network devices from several vendors can use a single RADIUS server-based security database.

- Networks already using RADIUS.

You can add a Cisco Nexus device with RADIUS to the network. This action might be the first step when you make a transition to an AAA server.

- Networks that require resource accounting.

You can use RADIUS accounting independent of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, indicating the amount of resources (such as time, packets, bytes, and so on) used during the session. An Internet service provider (ISP) might use a freeware-based version of the RADIUS access control and accounting software to meet special security and billing needs.

- Networks that support authentication profiles.

Using the RADIUS server in your network, you can configure AAA authentication and set up per-user profiles. Per-user profiles enable the Cisco Nexus device to manage ports using their existing RADIUS solutions and to efficiently manage shared resources to offer different service-level agreements.

Information About RADIUS Operations

When a user attempts to log in and authenticate to a Cisco Nexus device using RADIUS, the following process occurs:

1. The user is prompted for and enters 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 not authenticated and is prompted to reenter the username and password, or access is denied.
 - CHALLENGE—A challenge is issued by the RADIUS server. The challenge collects additional data from the user.
 - CHANGE PASSWORD—A request is issued by the RADIUS server, asking the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for EXEC or network authorization. You must first complete RADIUS authentication before using RADIUS authorization. The additional data included with the ACCEPT or REJECT packets consists of the following:

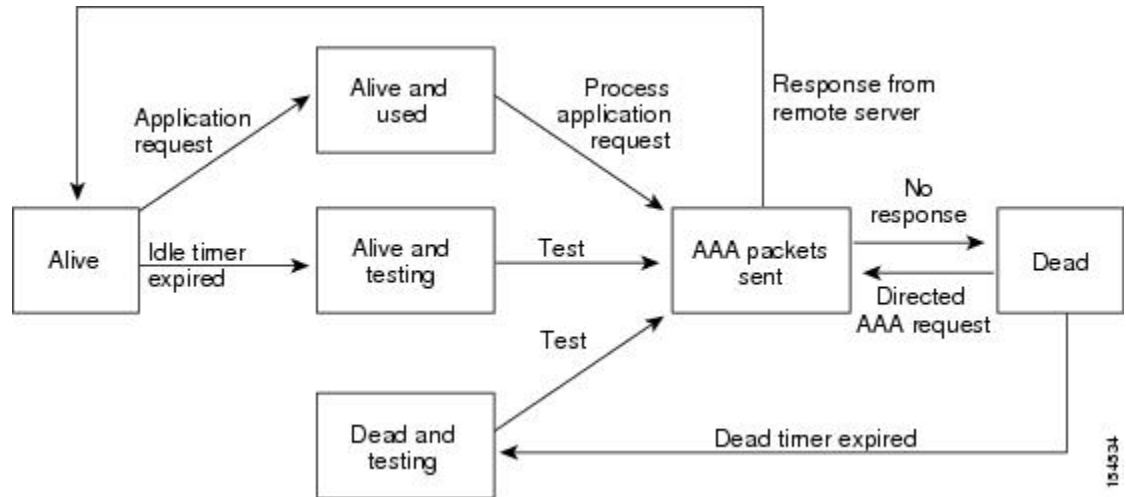
- Services that the user can access, including Telnet, rlogin, or local-area transport (LAT) connections, and Point-to-Point Protocol (PPP), Serial Line Internet Protocol (SLIP), or EXEC services.
- Connection parameters, including the host or client IPv4 or IPv6 address, access list, and user timeouts.

RADIUS Server Monitoring

An unresponsive RADIUS server can cause delay in processing of AAA requests. You can configure the switch to periodically monitor a RADIUS server to check whether it is responding (or alive) to save time in processing AAA requests. The switch marks unresponsive RADIUS servers as dead and does not send AAA requests to any dead RADIUS servers. The switch periodically monitors the dead RADIUS servers and brings them to the alive state once they respond. This process verifies that a RADIUS server is in a working state before real AAA requests are sent to the server. Whenever a RADIUS server changes to the dead or alive state, a Simple Network Management Protocol (SNMP) trap is generated and the switch displays an error message that a failure is taking place.

The following figure shows the different RADIUS server states:

Figure 2: RADIUS Server States



Note The monitoring interval for alive servers and dead servers are different and can be configured by the user. The RADIUS server monitoring is performed by sending a test authentication request to the RADIUS server.

Vendor-Specific Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific attributes (VSAs) between the network access server and the RADIUS server. The IETF uses attribute 26. VSAs allow vendors to support their own extended attributes that are not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option using the format recommended in the specification. The Cisco vendor ID is 9, and the supported option is vendor type 1, which is named cisco-av-pair. The value is a string with the following format:

```
protocol : attribute separator value *
```

The protocol is a Cisco attribute for a particular type of authorization, the separator is an equal sign (=) for mandatory attributes, and an asterisk (*) indicates optional attributes.

When you use RADIUS servers for authentication on a Cisco Nexus device, the RADIUS protocol directs the RADIUS server to return user attributes, such as authorization information, with authentication results. This authorization information is specified through VSAs.

The following VSA protocol options are supported by the Cisco Nexus device:

- Shell— Used in access-accept packets to provide user profile information.
- Accounting— Used in accounting-request packets. If a value contains any white spaces, you should enclose the value within double quotation marks.

The Cisco Nexus device supports the following attributes:

- roles—Lists all the roles to which the user belongs. The value field is a string that lists the role names delimited by white spaces.

- **accountinginfo**—Stores accounting information in addition to the attributes covered by a standard RADIUS accounting protocol. This attribute is sent only in the VSA portion of the Account-Request frames from the RADIUS client on the switch. It can be used only with the accounting protocol data units (PDUs).

Prerequisites for RADIUS

RADIUS has the following prerequisites:

- You must obtain IPv4 or IPv6 addresses or hostnames for the RADIUS servers.
- You must obtain preshared keys from the RADIUS servers.
- Ensure that the Cisco Nexus device is configured as a RADIUS client of the AAA servers.

Guidelines and Limitations for RADIUS

RADIUS has the following configuration guidelines and limitations:

- You can configure a maximum of 64 RADIUS servers on the Cisco Nexus device.

Configuring RADIUS Servers

This section describes how to configure RADIUS servers.

Procedure

- Step 1** Establish the RADIUS server connections to the Cisco Nexus device.
 - Step 2** Configure the preshared secret keys for the RADIUS servers.
 - Step 3** If needed, configure RADIUS server groups with subsets of the RADIUS servers for AAA authentication methods.
 - Step 4** If needed, configure any of the following optional parameters:
 - Dead-time interval.
 - Allow specification of a RADIUS server at login.
 - Transmission retry count and timeout interval.
 - Accounting and authentication attributes.
 - Step 5** If needed, configure periodic RADIUS server monitoring.
-

Configuring RADIUS Server Hosts

You must configure the IPv4 or IPv6 address or the hostname for each RADIUS server that you want to use for authentication. All RADIUS server hosts are added to the default RADIUS server group. You can configure up to 64 RADIUS servers.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> }	Specifies the IPv4 or IPv6 address or hostname for a RADIUS server.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show radius-server	Displays the RADIUS server configuration.
Step 5	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to configure host 10.10.1.1 as a RADIUS server:

```
switch# configure terminal
switch(config)# radius-server host 10.10.1.1
switch(config)# exit
switch# copy running-config startup-config
```

Configuring RADIUS Global Preshared Keys

You can configure preshared keys at the global level for all servers used by the Cisco Nexus device. A preshared key is a shared secret text string between the switch and the RADIUS server hosts.

Before you begin

Obtain the preshared key values for the remote RADIUS servers

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# radius-server key [0 7] <i>key-value</i>	Specifies a preshared key for all RADIUS servers. You can specify a clear text (0) or encrypted (7) preshared key. The default format is clear text. The maximum length is 63 characters. By default, no preshared key is configured.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show radius-server	Displays the RADIUS server configuration.

	Command or Action	Purpose
		Note The preshared keys are saved in encrypted form in the running configuration. Use the show running-config command to display the encrypted preshared keys.
Step 5	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure preshared keys at the global level for all servers used by the device:

```
switch# configure terminal
switch(config)# radius-server key 0 QsEfThUkO
switch(config)# exit
switch# copy running-config startup-config
```

Configuring RADIUS Server Preshared Keys

A preshared key is a shared secret text string between the Cisco Nexus device and the RADIUS server host.

Before you begin

Obtain the preshared key values for the remote RADIUS servers.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } key [0 7] <i>key-value</i>	Specifies a preshared key for a specific RADIUS server. You can specify a clear text (0) or encrypted (7) preshared key. The default format is clear text. The maximum length is 63 characters. This preshared key is used instead of the global preshared key.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show radius-server	Displays the RADIUS server configuration.

	Command or Action	Purpose
		Note The preshared keys are saved in encrypted form in the running configuration. Use the show running-config command to display the encrypted preshared keys.
Step 5	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure RADIUS preshared keys:

```
switch# configure terminal
switch(config)# radius-server host 10.10.1.1 key 0 PlIjUhYg
switch(config)# exit
switch# show radius-server
switch# copy running-config startup-config
```

Configuring RADIUS Server Groups

You can specify one or more remote AAA servers for authentication using server groups. All members of a group must belong to the RADIUS protocol. The servers are tried in the same order in which you configure them.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch (config)# aaa group server radius <i>group-name</i>	Creates a RADIUS server group and enters the RADIUS server group configuration submode for that group. The <i>group-name</i> argument is a case-sensitive, alphanumeric string with a maximum of 127 characters.
Step 3	switch (config-radius)# server { <i>ipv4-address</i> <i>ipv6-address</i> <i>server-name</i> }	Configures the RADIUS server as a member of the RADIUS server group. If the specified RADIUS server is not found, configure it using the radius-server host command and retry this command.
Step 4	(Optional) switch (config-radius)# deadtime <i>minutes</i>	Configures the monitoring dead time. The default is 0 minutes. The range is from 1 through 1440.

	Command or Action	Purpose
		Note If the dead-time interval for a RADIUS server group is greater than zero (0), that value takes precedence over the global dead-time value.
Step 5	(Optional) switch(config-radius)# source-interface <i>interface</i>	Assigns a source interface for a specific RADIUS server group. The supported interface types are management and VLAN. Note Use the source-interface command to override the global source interface assigned by the ip radius source-interface command.
Step 6	switch(config-radius)# exit	Exits configuration mode.
Step 7	(Optional) switch(config)# show radius-server group [<i>group-name</i>]	Displays the RADIUS server group configuration.
Step 8	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to configure a RADIUS server group:

```
switch# configure terminal
switch (config)# aaa group server radius RadServer
switch (config-radius)# server 10.10.1.1
switch (config-radius)# deadtime 30
switch (config-radius)# use-vrf management
switch (config-radius)# exit
switch (config)# show radius-server group
switch (config)# copy running-config startup-config
```

What to do next

Apply the RADIUS server groups to an AAA service.

Configuring the Global Source Interface for RADIUS Server Groups

You can configure a global source interface for RADIUS server groups to use when accessing RADIUS servers. You can also configure a different source interface for a specific RADIUS server group.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# ip radius source-interface interface	Configures the global source interface for all RADIUS server groups configured on the device. The source interface can be the management or the VLAN interface.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show radius-server	Displays the RADIUS server configuration information.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to configure the mgmt 0 interface as the global source interface for RADIUS server groups:

```
switch# configure terminal
switch(config)# ip radius source-interface mgmt 0
switch(config)# exit
switch# copy running-config startup-config
```

Allowing Users to Specify a RADIUS Server at Login

You can allow users to specify a RADIUS server at login.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# radius-server directed-request	Allows users to specify a RADIUS server to send the authentication request when logging in. The default is disabled.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show radius-server directed-request	Displays the directed request configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to allow users to select a RADIUS server when logging in to a network:

```
switch# configure terminal
switch(config)# radius-server directed-request
switch# exit
switch# copy running-config startup-config
```

Configuring the Global RADIUS Transmission Retry Count and Timeout Interval

You can configure a global retransmission retry count and timeout interval for all RADIUS servers. By default, a switch retries transmission to a RADIUS server only once before reverting to local authentication. You can increase this number up to a maximum of five retries per server. The timeout interval determines how long the Cisco Nexus device waits for responses from RADIUS servers before declaring a timeout failure.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# radius-server retransmit count	Specifies the retransmission count for all RADIUS servers. The default retransmission count is 1 and the range is from 0 to 5.
Step 3	switch(config)# radius-server timeout seconds	Specifies the transmission timeout interval for RADIUS servers. The default timeout interval is 5 seconds and the range is from 1 to 60 seconds.
Step 4	switch(config)# exit	Exits configuration mode.
Step 5	(Optional) switch# show radius-server	Displays the RADIUS server configuration.
Step 6	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to set the retry count to 3 and the transmission timeout interval to 5 seconds for RADIUS servers:

```
switch# configure terminal
switch(config)# radius-server retransmit 3
switch(config)# radius-server timeout 5
switch(config)# exit
switch# copy running-config startup-config
```

Configuring the RADIUS Transmission Retry Count and Timeout Interval for a Server

By default, a Cisco Nexus switch retries transmission to a RADIUS server only once before reverting to local authentication. You can increase this number up to a maximum of five retries per server. You can also set a timeout interval that the switch waits for responses from RADIUS servers before declaring a timeout failure.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } retransmit count	Specifies the retransmission count for a specific server. The default is the global value. Note The retransmission count value specified for a RADIUS server overrides the count specified for all RADIUS servers.
Step 3	switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } timeout seconds	Specifies the transmission timeout interval for a specific server. The default is the global value. Note The timeout interval value specified for a RADIUS server overrides the interval value specified for all RADIUS servers.
Step 4	switch(config)# exit	Exits configuration mode.
Step 5	(Optional) switch# show radius-server	Displays the RADIUS server configuration.
Step 6	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to set the RADIUS transmission retry count to 3 and the timeout interval to 10 seconds on RADIUS host server server1:

```
switch# configure terminal
switch(config)# radius-server host server1 retransmit 3
switch(config)# radius-server host server1 timeout 10
switch(config)# exit
switch# copy running-config startup-config
```

Configuring Accounting and Authentication Attributes for RADIUS Servers

You can specify that a RADIUS server is to be used only for accounting purposes or only for authentication purposes. By default, RADIUS servers are used for both accounting and authentication. You can also specify the destination UDP port numbers where RADIUS accounting and authentication messages should be sent.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	(Optional) switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } acct-port <i>udp-port</i>	Specifies a UDP port to use for RADIUS accounting messages. The default UDP port is 1812. The range is from 0 to 65535.
Step 3	(Optional) switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } accounting	Specifies that the specified RADIUS server is to be used only for accounting purposes. The default is both accounting and authentication.
Step 4	(Optional) switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } auth-port <i>udp-port</i>	Specifies a UDP port to use for RADIUS authentication messages. The default UDP port is 1812. The range is from 0 to 65535.
Step 5	(Optional) switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } authentication	Specifies that the specified RADIUS server only be used for authentication purposes. The default is both accounting and authentication.
Step 6	switch(config)# exit	Exits configuration mode.
Step 7	(Optional) switch(config)# show radius-server	Displays the RADIUS server configuration.
Step 8	switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure accounting and authentication attributes for a RADIUS server:

```
switch# configure terminal
switch(config)# radius-server host 10.10.1.1 acct-port 2004
switch(config)# radius-server host 10.10.1.1 accounting
switch(config)# radius-server host 10.10.2.2 auth-port 2005
switch(config)# radius-server host 10.10.2.2 authentication
switch # exit
switch # copy running-config startup-config
switch #
```

Configuring Periodic RADIUS Server Monitoring

You can monitor the availability of RADIUS servers. These parameters include the username and password to use for the server and an idle timer. The idle timer specifies the interval during which a RADIUS server receives no requests before the switch sends out a test packet. You can configure this option to test servers periodically.



Note For security reasons, we recommend that you do not configure a test username that is the same as an existing user in the RADIUS database.

The test idle timer specifies the interval during which a RADIUS server receives no requests before the switch sends out a test packet.

The default idle timer value is 0 minutes. When the idle time interval is 0 minutes, the switch does not perform periodic RADIUS server monitoring.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } test { idle-time <i>minutes</i> password <i>password</i> [<i>idle-time minutes</i>] username <i>name</i> [password password [<i>idle-time minutes</i>]]}	Specifies parameters for server monitoring. The default username is test and the default password is test. The default value for the idle timer is 0 minutes. The valid range is from 0 to 1440 minutes. Note For periodic RADIUS server monitoring, you must set the idle timer to a value greater than 0.
Step 3	switch(config)# radius-server deadtime <i>minutes</i>	Specifies the number of minutes before the switch checks a RADIUS server that was previously unresponsive. The default value is 0 minutes. The valid range is 1 to 1440 minutes.
Step 4	switch(config)# exit	Exits configuration mode.
Step 5	(Optional) switch# show radius-server	Displays the RADIUS server configuration.
Step 6	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure RADIUS server host 10.10.1.1 with a username (user1) and password (Ur2Gd2BH) and with an idle timer of 3 minutes and a deadtime of 5 minutes:

```
switch# configure terminal
switch(config)# radius-server host 10.10.1.1 test username user1 password Ur2Gd2BH idle-time
3
switch(config)# radius-server deadtime 5
switch(config)# exit
switch# copy running-config startup-config
```

Configuring the Dead-Time Interval

You can configure the dead-time interval for all RADIUS servers. The dead-time interval specifies the time that the Cisco Nexus device waits after declaring a RADIUS server is dead, before sending out a test packet to determine if the server is now alive. The default value is 0 minutes.



Note When the dead-time interval is 0 minutes, RADIUS servers are not marked as dead even if they are not responding. You can configure the dead-time interval for a RADIUS server group.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# radius-server deadtime	Configures the dead-time interval. The default value is 0 minutes. The range is from 1 to 1440 minutes.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show radius-server	Displays the RADIUS server configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to configure a deadtime of 5 minutes for a radius server:

```
switch# configure terminal
switch(config)# radius-server deadtime 5
switch(config)# exit
switch# copy running-config startup-config
```

Manually Monitoring RADIUS Servers or Groups

Procedure

	Command or Action	Purpose
Step 1	switch# test aaa server radius {ipv4-address ipv6-address server-name} [vrf vrf-name] username password test aaa server radius {ipv4-address ipv6-address server-name} [vrf vrf-name] username password	Sends a test message to a RADIUS server to confirm availability.
Step 2	switch# test aaa group group-name username password	Sends a test message to a RADIUS server group to confirm availability.

Example

This example shows how to send a test message to the RADIUS server and server group to confirm availability:

```
switch# test aaa server radius 10.10.1.1 user 1 Ur2Gd2BH
switch# test aaa group RadGroup user2 As3He3CI
```

Verifying the RADIUS Configuration

Procedure

	Command or Action	Purpose
Step 1	switch# show running-config radius [all]	Displays the RADIUS configuration in the running configuration.
Step 2	switch# show startup-config radius	Displays the RADIUS configuration in the startup configuration.
Step 3	switch# show radius-server [<i>server-name</i> <i>ipv4-address</i> <i>ipv6-address</i>] [directed-request groups sorted statistics]	Displays all configured RADIUS server parameters.

Displaying RADIUS Server Statistics

Procedure

	Command or Action	Purpose
Step 1	switch# show radius-server statistics { <i>hostname</i> <i>ipv4-address</i> <i>ipv6-address</i> }	Displays the RADIUS statistics.

Clearing RADIUS Server Statistics

You can display the statistics that the Cisco NX-OS device maintains for RADIUS server activity.

Before you begin

Configure RADIUS servers on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	(Optional) switch# show radius-server statistics { <i>hostname</i> <i>ipv4-address</i> <i>ipv6-address</i> }	Displays the RADIUS server statistics on the Cisco NX-OS device.

	Command or Action	Purpose
Step 2	switch# clear radius-server statistics {hostname ipv4-address ipv6-address}	Clears the RADIUS server statistics.

Configuration Examples for RADIUS

The following example shows how to configure RADIUS:

```
switch# configure terminal
switch(config)# radius-server key 7 "ToIkLhPg"
switch(config)# radius-server host 10.10.1.1 key 7 "ShMoMhTl" authentication accounting
switch(config)# aaa group server radius RadServer
switch(config-radius)# server 10.10.1.1
switch(config-radius)# exit
switch(config-radius)# use-vrf management
```

Default Settings for RADIUS

The following table lists the default settings for RADIUS parameters.

Table 6: Default RADIUS Parameters

Parameters	Default
Server roles	Authentication and accounting
Dead timer interval	0 minutes
Retransmission count	1
Retransmission timer interval	5 seconds
Idle timer interval	0 minutes
Periodic server monitoring username	test
Periodic server monitoring password	test



CHAPTER 5

Configuring TACACS+

This chapter contains the following sections:

- [About Configuring TACACS+, on page 37](#)

About Configuring TACACS+

Information About TACACS+

The Terminal Access Controller Access Control System Plus (TACACS+) security protocol provides centralized validation of users attempting to gain access to a Cisco Nexus device. TACACS+ services are maintained in a database on a TACACS+ daemon typically running on a UNIX or Windows NT workstation. You must have access to and must configure a TACACS+ server before the configured TACACS+ features on your Cisco Nexus device are available.

TACACS+ provides for separate 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 is associated with 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 TACACS+ client/server protocol uses TCP (TCP port 49) for transport requirements. The Cisco Nexus device provides centralized authentication using the TACACS+ protocol.

TACACS+ Advantages

TACACS+ has the following advantages over RADIUS authentication:

- Provides independent AAA facilities. For example, the Cisco Nexus device can authorize access without authenticating.
- Uses the TCP transport protocol to send data between the AAA client and server, making reliable transfers with a connection-oriented protocol.
- Encrypts the entire protocol payload between the switch and the AAA server to ensure higher data confidentiality. The RADIUS protocol only encrypts passwords.

User Login with TACACS+

When a user attempts a Password Authentication Protocol (PAP) login to a Cisco Nexus device using TACACS+, the following actions occur:

1. When the Cisco Nexus device establishes a connection, it contacts the TACACS+ daemon to obtain the username and password.



Note

TACACS+ allows an arbitrary conversation between the daemon and the user until the daemon receives enough information to authenticate the user. This action is usually done by prompting for a username and password combination, but may include prompts for other items, such as the user's mother's maiden name.

2. The Cisco Nexus device receives one of the following responses from the TACACS+ daemon:
 - ACCEPT—User authentication succeeds and service begins. If the Cisco Nexus device requires user authorization, authorization begins.
 - REJECT—User authentication failed. The TACACS+ daemon either denies further access to the user or prompts the user to retry the login sequence.
 - ERROR—An error occurred at some time during authentication either at the daemon or in the network connection between the daemon and the Cisco Nexus device. If the Cisco Nexus device receives an ERROR response, the switch tries to use an alternative method for authenticating the user.

The user also undergoes an additional authorization phase, if authorization has been enabled on the Cisco Nexus device. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the Cisco Nexus device again contacts the TACACS+ daemon and it returns an ACCEPT or REJECT authorization response. An ACCEPT response contains attributes that are used to direct the EXEC or NETWORK session for that user and determines the services that the user can access.

Services include the following:

- Telnet, rlogin, Point-to-Point Protocol (PPP), Serial Line Internet Protocol (SLIP), or EXEC services
- Connection parameters, including the host or client IP address (IPv4 or IPv6), access list, and user timeouts

Default TACACS+ Server Encryption Type and Preshared Key

You must configure the TACACS+ preshared key to authenticate the switch to the TACACS+ server. A preshared key is a secret text string shared between the Cisco Nexus device and the TACACS+ server host. The length of the key is restricted to 63 characters and can include any printable ASCII characters (white spaces are not allowed). You can configure a global preshared secret key for all TACACS+ server configurations on the Cisco Nexus device to use.

You can override the global preshared key assignment by using the **key** option when configuring an individual TACACS+ server.

Command Authorization Support for TACACS+ Servers

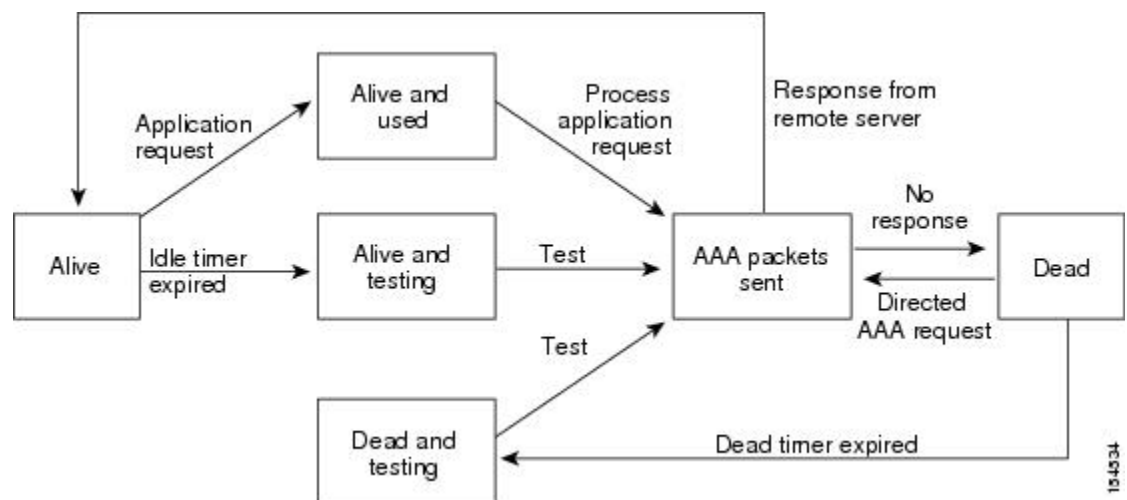
By default, command authorization is done against a local database in the Cisco NX-OS software when an authenticated user enters a command at the command-line interface (CLI). You can also verify authorized commands for authenticated users using TACACS+.

TACACS+ Server Monitoring

An unresponsive TACACS+ server can delay the processing of AAA requests. A Cisco Nexus device can periodically monitor an TACACS+ server to check whether it is responding (or alive) to save time in processing AAA requests. The Cisco Nexus device marks unresponsive TACACS+ servers as dead and does not send AAA requests to any dead TACACS+ servers. The Cisco Nexus device periodically monitors dead TACACS+ servers and brings them to the alive state once they are responding. This process verifies that a TACACS+ server is in a working state before real AAA requests are sent to the server. Whenever an TACACS+ server changes to the dead or alive state, a Simple Network Management Protocol (SNMP) trap is generated and the Cisco Nexus device displays an error message that a failure is taking place before it can impact performance.

The following figure shows the different TACACS+ server states:

Figure 3: TACACS+ Server States



Note The monitoring interval for alive servers and dead servers are different and can be configured by the user. The TACACS+ server monitoring is performed by sending a test authentication request to the TACACS+ server.

Prerequisites for TACACS+

TACACS+ has the following prerequisites:

- You must obtain the IPv4 or IPv6 addresses or hostnames for the TACACS+ servers.
- You must obtain the preshared keys from the TACACS+ servers, if any.
- Ensure that the Cisco Nexus device is configured as a TACACS+ client of the AAA servers.

Guidelines and Limitations for TACACS+

TACACS+ has the following configuration guidelines and limitations:

- You can configure a maximum of 64 TACACS+ servers on the Cisco Nexus device.

Configuring TACACS+

TACACS+ Server Configuration Process

This section describes how to configure TACACS+ servers.

Procedure

-
- Step 1** Enable TACACS+.
- Step 2** Establish the TACACS+ server connections to the Cisco Nexus device.
- Step 3** Configure the preshared secret keys for the TACACS+ servers.
- Step 4** If needed, configure TACACS+ server groups with subsets of the TACACS+ servers for AAA authentication methods.
- Step 5** If needed, configure any of the following optional parameters:
- Dead-time interval
 - Allow TACACS+ server specification at login
 - Timeout interval
 - TCP port
- Step 6** If needed, configure periodic TACACS+ server monitoring.
-

Enabling TACACS+

Although by default, the TACACS+ feature is disabled on the Cisco Nexus device. You can enable the TACACS+ feature to access the configuration and verification commands for authentication, perform this task:

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# feature tacacs+	Enables TACACS+.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring TACACS+ Server Hosts

To access a remote TACACS+ server, you must configure the IPv4 or IPv6 address or the hostname for the TACACS+ server on the Cisco Nexus device. All TACACS+ server hosts are added to the default TACACS+ server group. You can configure up to 64 TACACS+ servers.

If a preshared key is not configured for a configured TACACS+ server, a warning message is issued if a global key is not configured. If a TACACS+ server key is not configured, the global key (if configured) is used for that server.

Before you configure TACACS+ server hosts, you should do the following:

- Enable TACACS+.
- Obtain the IPv4 or IPv6 addresses or the hostnames for the remote TACACS+ servers.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# tacacs-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> }	Specifies the IPv4 or IPv6 address or hostname for a TACACS+ server.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show tacacs-server	Displays the TACACS+ server configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

You can delete a TACACS+ server host from a server group.

Configuring TACACS+ Global Preshared Keys

You can configure preshared keys at the global level for all servers used by the Cisco Nexus device. A preshared key is a shared secret text string between the Cisco Nexus device and the TACACS+ server hosts.

Before you configure preshared keys, you should do the following:

- Enable TACACS+.
- Obtain the preshared key values for the remote TACACS+ servers.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# tacacs-server key [0 7] <i>key-value</i>	Specifies a preshared key for all TACACS+ servers. You can specify a clear text (0) or encrypted (7) preshared key. The default

	Command or Action	Purpose
		format is clear text. The maximum length is 63 characters. By default, no preshared key is configured.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show tacacs-server	Displays the TACACS+ server configuration. Note The preshared keys are saved in encrypted form in the running configuration. Use the show running-config command to display the encrypted preshared keys.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to configure global preshared keys:

```
switch# configure terminal
switch(config)# tacacs-server key 0 QsEfThUkO
switch(config)# exit
switch# show tacacs-server
switch# copy running-config startup-config
```

Related Topics

[Enabling TACACS+](#) , on page 40

Configuring TACACS+ Server Preshared Keys

You can configure preshared keys for a TACACS+ server. A preshared key is a shared secret text string between the Cisco Nexus device and the TACACS+ server host.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# tacacs-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } key [0 7] <i>key-value</i>	Specifies a preshared key for a specific TACACS+ server. You can specify a clear text (0) or encrypted (7) preshared key. The default format is clear text. The maximum length is 63 characters. This preshared key is used instead of the global preshared key.
Step 3	switch(config)# exit	Exits configuration mode.

	Command or Action	Purpose
Step 4	(Optional) switch# show tacacs-server	Displays the TACACS+ server configuration. Note The preshared keys are saved in encrypted form in the running configuration. Use the show running-config command to display the encrypted preshared keys.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to configure the TACACS+ preshared keys:

```
switch# configure terminal
switch(config)# tacacs-server host 10.10.1.1 key 0 PlIjUhYg
switch(config)# exit
switch# show tacacs-server
switch# copy running-config startup-config
```

Configuring TACACS+ Server Groups

You can specify one or more remote AAA servers to authenticate users using server groups. All members of a group must belong to the TACACS+ protocol. The servers are tried in the same order in which you configure them.

You can configure these server groups at any time but they only take effect when you apply them to an AAA service.

Before you begin

You must use the **feature tacacs+** command to enable TACACS+ before you configure TACACS+.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# aaa group server tacacs+ group-name	Creates a TACACS+ server group and enters the TACACS+ server group configuration mode for that group.
Step 3	switch(config-tacacs+)# server {ipv4-address ipv6-address host-name}	Configures the TACACS+ server as a member of the TACACS+ server group. If the specified TACACS+ server is not found, configure it using the tacacs-server host command and retry this command.

	Command or Action	Purpose
Step 4	(Optional) switch(config-tacacs+)# deadtime <i>minutes</i>	Configures the monitoring dead time. The default is 0 minutes. The range is from 0 through 1440. Note If the dead-time interval for a TACACS+ server group is greater than zero (0), that value takes precedence over the global dead-time value.
Step 5	(Optional) switch(config-tacacs+)# source-interface <i>interface</i> Example: switch(config-tacacs+)# source-interface mgmt 0	Assigns a source interface for a specific TACACS+ server group. The supported interface types are management and VLAN. Note Use the source-interface command to override the global source interface assigned by the ip tacacs source-interface command.
Step 6	switch(config-tacacs+)# exit	Exits configuration mode.
Step 7	(Optional) switch(config)# show tacacs-server groups	Displays the TACACS+ server group configuration.
Step 8	(Optional) switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to configure a TACACS+ server group:

```
switch# configure terminal
switch(config)# aaa group server tacacs+ TacServer
switch(config-tacacs+)# server 10.10.2.2
switch(config-tacacs+)# deadtime 30
switch(config-tacacs+)# exit
switch(config)# show tacacs-server groups
switch(config)# copy running-config startup-config
```

Configuring the Global Source Interface for TACACS+ Server Groups

You can configure a global source interface for TACACS+ server groups to use when accessing TACACS+ servers. You can also configure a different source interface for a specific TACACS+ server group.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)	Enters global configuration mode.
Step 2	ip tacacs source-interface <i>interface</i> Example: switch(config)# ip tacacs source-interface mgmt 0	Configures the global source interface for all TACACS+ server groups configured on the device. The source interface can be the management or the VLAN interface.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) show tacacs-server Example: switch# show tacacs-server	Displays the TACACS+ server configuration information.
Step 5	(Optional) copy running-config startup config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Specifying a TACACS+ Server at Login

You can configure the switch to allow the user to specify which TACACS+ server to send the authenticate request by enabling the directed-request option. By default, a Cisco Nexus device forwards an authentication request based on the default AAA authentication method. If you enable this option, the user can log in as *username@hostname*, where *hostname* is the name of a configured RADIUS server.



Note User specified logins are only supported for Telnet sessions.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# tacacs-server directed-request	Allows users to specify a TACACS+ server to send the authentication request when logging in. The default is disabled.
Step 3	switch(config)# exit	Exits configuration mode.

	Command or Action	Purpose
Step 4	(Optional) <code>switch# show tacacs-server directed-request</code>	Displays the TACACS+ directed request configuration.
Step 5	(Optional) <code>switch# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Configuring AAA Authorization on TACACS+ Servers

You can configure the default AAA authorization method for TACACS+ servers.

Before you begin

Enable TACACS+.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	aaa authorization ssh-certificate default {group <i>group-list</i> [none] local none} Example: <pre>switch(config)# aaa authorization ssh-certificate default group TACACSServer1 TACACSServer2</pre>	<p>Configures the default AAA authorization method for the TACACS+ servers.</p> <p>The ssh-certificate keyword configures TACACS+ or local authorization with certificate authentication. The default authorization is local authorization, which is the list of authorized commands for the user's assigned role.</p> <p>The <i>group-list</i> argument consists of a space-delimited list of TACACS+ server group names. Servers belonging to this group are contacted for AAA authorization. The local method uses the local database for authorization, and the none method specifies that no AAA authorization be used.</p>
Step 3	exit Example: <pre>switch(config)# exit switch#</pre>	Exits global configuration mode.
Step 4	(Optional) show aaa authorization [all] Example: <pre>switch# show aaa authorization</pre>	Displays the AAA authorization configuration. The all keyword displays the default values.

	Command or Action	Purpose
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring Command Authorization on TACACS+ Servers

You can configure authorization for commands on TACACS+ servers. Command authorization disables user role-based authorization control (RBAC), including the default roles.



Note By default, context-sensitive help and command tab completion show only the commands that are supported for a user as defined by the assigned roles. When you enable command authorization, the Cisco NX-OS software displays all commands in the context sensitive help and in tab completion, regardless of the role assigned to the user.

Before you begin

Enable TACACS+.

Configure TACACS host and server groups before configuring AAA command authorization.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	aaa authorization {commands config-commands} default [group group-list [local] local] Example: switch(config)# aaa authorization commands default group TacGroup	Configures the default authorization method for commands for all roles. The commands keyword configures authorization sources for all EXEC commands, and the config-commands keyword configures authorization sources for all configuration commands. The default authorization for all commands is local authorization, which is the list of authorized commands for the user's assigned role. The <i>group-list</i> argument consists of a space-delimited list of TACACS+ server group names. Servers that belong to this group are contacted for command authorization. The local method uses the local role-based database for authorization.

	Command or Action	Purpose
		<p>The local method is used only if all the configured server groups fail to respond and you have configured local as the fallback method.</p> <p>The default method is local.</p> <p>If you have not configured a fallback method after the TACACS+ server group method, authorization fails if all server groups fail to respond.</p>
Step 3	exit Example: <pre>switch(config)# exit switch#</pre>	Exits global configuration mode.
Step 4	(Optional) show aaa authorization [all] Example: <pre>switch(config)# show aaa authorization</pre>	Displays the AAA authorization configuration. The all keyword displays the default values.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Testing Command Authorization on TACACS+ Servers

You can test the command authorization for a user on the TACACS+ servers.



Note You must send correct commands for authorization or the results might not be reliable.

Before you begin

Enable TACACS+.

Ensure that you have configured command authorization for the TACACS+ servers.

Procedure

	Command or Action	Purpose
Step 1	test aaa authorization command-type {commands config-commands} user username command command-string Example:	<p>Tests a user's authorization for a command on the TACACS+ servers.</p> <p>The commands keyword specifies only EXEC commands and the config-commands keyword specifies only configuration commands.</p>

	Command or Action	Purpose
	<pre>switch# test aaa authorization command-type commands user TestUser command reload</pre>	Note Put double quotes (") before and after the <i>command-string</i> argument if it contains spaces.

Enabling and Disabling Command Authorization Verification

You can enable and disable command authorization verification on the command-line interface (CLI) for the default user session or for another username.



Note The commands do not execute when you enable authorization verification.

Procedure

	Command or Action	Purpose
Step 1	terminal verify-only [username <i>username</i>] Example: <pre>switch# terminal verify-only</pre>	Enables command authorization verification. After you enter this command, the Cisco NX-OS software indicates whether the commands you enter are authorized or not.
Step 2	terminal no verify-only [username <i>username</i>] Example: <pre>switch# terminal no verify-only</pre>	Disables command authorization verification.

Configuring Privilege Level Support for Authorization on TACACS+ Servers

You can configure privilege level support for authorization on TACACS+ servers.

Unlike Cisco IOS devices, which use privilege levels to determine authorization, Cisco NX-OS devices use role-based access control (RBAC). To enable both types of devices to be administered by the same TACACS+ servers, you can map the privilege levels configured on TACACS+ servers to user roles configured on Cisco NX-OS devices.

When a user authenticates with a TACACS+ server, the privilege level is obtained and used to form a local user role name of the format “priv-*n*,” where *n* is the privilege level. The user assumes the permissions of this local role. Sixteen privilege levels, which map directly to corresponding user roles, are available. The following table shows the user role permissions that correspond to each privilege level.

Privilege Level	User Role Permissions
15	network-admin permissions
14	vdc-admin permissions
13 - 1	<ul style="list-style-type: none"> Standalone role permissions, if the feature privilege command is disabled. Same permissions as privilege level 0 with cumulative privileges for roles, if the feature privilege command is enabled.

Privilege Level	User Role Permissions
0	Permission to execute show commands and exec commands (such as ping , trace , and ssh).



Note When the **feature privilege** command is enabled, privilege roles inherit the permissions of lower level privilege roles.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] feature privilege Example: <pre>switch(config)# feature privilege</pre>	Enables or disables the cumulative privilege of roles. Users can see the enable command only if this feature is enabled. The default is disabled.
Step 3	[no] enable secret [0 5] password [priv-lvl priv-lvl all] Example: <pre>switch(config)# enable secret 5 def456 priv-lvl 15</pre>	<p>Enables or disables a secret password for a specific privilege level. Users are prompted to enter the correct password upon each privilege level escalation. The default is disabled.</p> <p>You can enter 0 to specify that the password is in clear text or 5 to specify that the password is in encrypted format. The <i>password</i> argument can be up to 64 alphanumeric characters. The <i>priv-lvl</i> argument is from 1 to 15.</p> <p>Note To enable the secret password, you must have enabled the cumulative privilege of roles by entering the feature privilege command.</p>
Step 4	[no] username username priv-lvl n Example: <pre>switch(config)# username user2 priv-lvl 15</pre>	<p>Enables or disables a user to use privilege levels for authorization. The default is disabled.</p> <p>The priv-lvl keyword specifies the privilege level to which the user is assigned. There is no default privilege level. Privilege levels 0 to 15 (priv-lvl 0 to priv-lvl 15) map to user roles priv-0 to priv-15.</p>
Step 5	(Optional) show privilege Example: <pre>switch(config)# show privilege</pre>	Displays the username, current privilege level, and status of cumulative privilege support.

	Command or Action	Purpose
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.
Step 7	exit Example: <pre>switch(config)# exit switch#</pre>	Exits global configuration mode.
Step 8	enable level Example: <pre>switch# enable 15</pre>	Enables a user to move to a higher privilege level. This command prompts for the secret password. The <i>level</i> argument specifies the privilege level to which the user is granted access. The only available level is 15.

Permitting or Denying Commands for Users of Privilege Roles

As a network administrator, you can modify the privilege roles to permit users to execute specific commands or to prevent users from running those commands.

You must follow these guidelines when changing the rules of privilege roles:

- You cannot modify the priv-14 and priv-15 roles.
- You can add deny rules only to the priv-0 role.
- These commands are always permitted for the priv-0 role: **configure**, **copy**, **dir**, **enable**, **ping**, **show**, **ssh**, **telnet**, **terminal**, **traceroute**, **end**, and **exit**.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] role name priv-n Example: <pre>switch(config)# role name priv-5 switch(config-role)#</pre>	Enables or disables a privilege role and enters role configuration mode. The <i>n</i> argument specifies the privilege level and is a number between 0 and 13.
Step 3	rule number {deny permit} command command-string Example: <pre>switch(config-role)# rule 2 permit command pwd</pre>	Configures a command rule for users of privilege roles. These rules permit or deny users to execute specific commands. You can configure up to 256 rules for each role. The rule number determines the order in which the rules

	Command or Action	Purpose
		are applied. Rules are applied in descending order. For example, if a role has three rules, rule 3 is applied before rule 2, which is applied before rule 1. The <i>command-string</i> argument can contain spaces. Note Repeat this command for 256 rules.
Step 4	exit Example: <pre>switch(config-role)# exit switch(config)#</pre>	Exits role configuration mode.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the Global TACACS+ Timeout Interval

You can set a global timeout interval that the Cisco Nexus device waits for responses from all TACACS+ servers before declaring a timeout failure. The timeout interval determines how long the switch waits for responses from TACACS+ servers before declaring a timeout failure.

Procedure

	Command or Action	Purpose
Step 1	<code>switch# configure terminal</code>	Enters configuration mode.
Step 2	<code>switch(config)# tacacs-server timeout <i>seconds</i></code>	Specifies the timeout interval for TACACS+ servers. The default timeout interval is 5 second and the range is from 1 to 60 seconds.
Step 3	<code>switch(config)# exit</code>	Exits configuration mode.
Step 4	(Optional) <code>switch# show tacacs-server</code>	Displays the TACACS+ server configuration.
Step 5	(Optional) <code>switch# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Configuring the Timeout Interval for a Server

You can set a timeout interval that the Cisco Nexus device waits for responses from a TACACS+ server before declaring a timeout failure. The timeout interval determines how long the switch waits for responses from a TACACS+ server before declaring a timeout failure.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# switch(config)# tacacs-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } timeout <i>seconds</i>	Specifies the timeout interval for a specific server. The default is the global value. Note The timeout interval value specified for a TACACS+ server overrides the global timeout interval value specified for all TACACS+ servers.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show tacacs-server	Displays the TACACS+ server configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring TCP Ports

You can configure another TCP port for the TACACS+ servers if there are conflicts with another application. By default, the Cisco Nexus device uses port 49 for all TACACS+ requests.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# tacacs-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } port <i>tcp-port</i>	Specifies the UDP port to use for TACACS+ accounting messages. The default TCP port is 49. The range is from 1 to 65535.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show tacacs-server	Displays the TACACS+ server configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to configure TCP ports:

```
switch# configure terminal
switch(config)# tacacs-server host 10.10.1.1 port 2
switch(config)# exit
switch# show tacacs-server
switch# copy running-config startup-config
```

Configuring Periodic TACACS+ Server Monitoring

You can monitor the availability of TACACS+ servers. These parameters include the username and password to use for the server and an idle timer. The idle timer specifies the interval in which a TACACS+ server receives no requests before the Cisco Nexus device sends out a test packet. You can configure this option to test servers periodically, or you can run a one-time only test.



Note To protect network security, we recommend that you use a username that is not the same as an existing username in the TACACS+ database.

The test idle timer specifies the interval in which a TACACS+ server receives no requests before the Cisco Nexus device sends out a test packet.



Note The default idle timer value is 0 minutes. When the idle time interval is 0 minutes, periodic TACACS+ server monitoring is not performed.

To configure periodic TACACS+ server monitoring, perform this task:

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# tacacs-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } test { <i>idle-time minutes</i> password password [<i>idle-time minutes</i>] username name [password password [<i>idle-time minutes</i>]]}	Specifies parameters for server monitoring. The default username is test and the default password is test. The default value for the idle timer is 0 minutes and the valid range is from 0 to 1440 minutes. Note For periodic TACACS+ server monitoring, the idle timer value must be greater than 0.
Step 3	switch(config)# tacacs-server dead-time <i>minutes</i>	Specifies the number minutes before the Cisco Nexus device checks a TACACS+ server that was previously unresponsive. The default value is 0 minutes and the valid range is 0 to 1440 minutes.
Step 4	switch(config)# exit	Exits configuration mode.
Step 5	(Optional) switch# show tacacs-server	Displays the TACACS+ server configuration.
Step 6	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to configure periodic TACACS+ server monitoring:

```
switch# configure terminal
switch(config)# tacacs-server host 10.10.1.1 test username user1 password Ur2Gd2BH idle-time
3
switch(config)# tacacs-server dead-time 5
switch(config)# exit
switch# show tacacs-server
switch# copy running-config startup-config
```

Configuring the Dead-Time Interval

You can configure the dead-time interval for all TACACS+ servers. The dead-time interval specifies the time that the Cisco Nexus device waits, after declaring a TACACS+ server is dead, before sending out a test packet to determine if the server is now alive.



Note When the dead-time interval is 0 minutes, TACACS+ servers are not marked as dead even if they are not responding. You can configure the dead-time interval per group.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# tacacs-server deadtime <i>minutes</i>	Configures the global dead-time interval. The default value is 0 minutes. The range is from 1 to 1440 minutes.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# show tacacs-server	Displays the TACACS+ server configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Manually Monitoring TACACS+ Servers or Groups

Procedure

	Command or Action	Purpose
Step 1	switch# test aaa server tacacs+ { <i>ipv4-address</i> <i>ipv6-address</i> <i>host-name</i> } [vrf <i>vrf-name</i>] <i>username password</i>	Sends a test message to a TACACS+ server to confirm availability.
Step 2	switch# test aaa group <i>group-name username</i> <i>password</i>	Sends a test message to a TACACS+ server group to confirm availability.

Example

The following example shows how to manually issue a test message:

```
switch# test aaa server tacacs+ 10.10.1.1 user1 Ur2Gd2BH
switch# test aaa group TacGroup user2 As3He3CI
```

Disabling TACACS+

You can disable TACACS+.

**Caution**

When you disable TACACS+, all related configurations are automatically discarded.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# no feature tacacs+	Disables TACACS+.
Step 3	switch(config)# exit	Exits configuration mode.
Step 4	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Displaying TACACS+ Statistics

To display the statistics, the switch maintains for TACACS+ activity, perform this task:

Procedure

	Command or Action	Purpose
Step 1	switch# show tacacs-server statistics {hostname ipv4-address ipv6-address}	Displays the TACACS+ statistics.

Example

For detailed information about the fields in the output from this command, see the *Command Reference* for your Nexus switch.

Verifying the TACACS+ Configuration

To display TACACS+ configuration information, perform one of the following tasks:

Procedure

	Command or Action	Purpose
Step 1	switch# show tacacs+ {status pending pending-diff}	Displays the TACACS+ Cisco Fabric Services distribution status and other details.
Step 2	switch# show running-config tacacs [all]	Displays the TACACS+ configuration in the running configuration.
Step 3	switch# show startup-config tacacs	Displays the TACACS+ configuration in the startup configuration.
Step 4	switch# show tacacs-serve [host-name ipv4-address ipv6-address] [directed-request groups sorted statistics]	Displays all configured TACACS+ server parameters.

Configuration Examples for TACACS+

The following example shows how to configure TACACS+:

```
switch# configure terminal
switch(config)# feature tacacs+
switch(config)# tacacs-server key 7 "ToIkLhPpG"
switch(config)# tacacs-server host 10.10.2.2 key 7 "ShMoMhTl"
switch(config)# aaa group server tacacs+ TacServer
switch(config-tacacs+)# server 10.10.2.2
switch(config-tacacs+)# use-vrf management
```

The following example shows how to enable tacacs+ and how to configure the tacacs+ server preshared keys to specify remote AAA servers to authenticate server group TacServer1:

```
switch# config t
switch(config)# feature tacacs+
switch(config)# tacacs-server key 7 "ikvhwl0"
switch(config)# tacacs-server host 1.1.1.1
switch(config)# tacacs-server host 1.1.1.2

switch(config)# aaa group server tacacs+ TacServer1
switch(config-tacacs+)# server 1.1.1.1
switch(config-tacacs+)# server 1.1.1.2
```

Default Settings for TACACS+

The following table lists the default settings for TACACS+ parameters.

Table 7: Default TACACS+ Parameters

Parameters	Default
TACACS+	Disabled
Dead-time interval	0 minutes

Parameters	Default
Timeout interval	5 seconds
Idle timer interval	0 minutes
Periodic server monitoring username	test
Periodic server monitoring password	test



CHAPTER 6

Configuring SSH and Telnet

This chapter contains the following sections:

- [Configuring SSH and Telnet, on page 59](#)

Configuring SSH and Telnet

Information About SSH and Telnet

SSH Server

The Secure Shell Protocol (SSH) server feature enables a SSH client to make a secure, encrypted connection to a Cisco Nexus device. SSH uses strong encryption for authentication. The SSH server in the Cisco Nexus device switch interoperates with publicly and commercially available SSH clients.

The user authentication mechanisms supported for SSH are RADIUS, TACACS+, and the use of locally stored user names and passwords.

SSH Client

The SSH client feature is an application running over the SSH protocol to provide device authentication and encryption. The SSH client enables a switch to make a secure, encrypted connection to another Cisco Nexus device or to any other device running an SSH server. This connection provides an outbound connection that is encrypted. With authentication and encryption, the SSH client allows for a secure communication over an insecure network.

The SSH client in the Cisco Nexus device works with publicly and commercially available SSH servers.

SSH Server Keys

SSH requires server keys for secure communications to the Cisco Nexus device. You can use SSH keys for the following SSH options:

- SSH version 2 using Rivest, Shamir, and Adelman (RSA) public-key cryptography
- SSH version 2 using the Digital System Algorithm (DSA)

Be sure to have an SSH server key-pair with the appropriate version before enabling the SSH service. You can generate the SSH server key-pair according to the SSH client version used. The SSH service accepts three types of key-pairs for use by SSH version 2:

- The `dsa` option generates the DSA key-pair for the SSH version 2 protocol.
- The `rsa` option generates the RSA key-pair for the SSH version 2 protocol.

By default, the Cisco Nexus device generates an RSA key using 1024 bits.

SSH supports the following public key formats:

- OpenSSH
- IETF Secure Shell (SECSH)


Caution

If you delete all of the SSH keys, you cannot start the SSH services.

Telnet Server

The Telnet protocol enables TCP/IP connections to a host. Telnet allows a user at one site to establish a TCP connection to a login server at another site, and then passes the keystrokes from one system to the other. Telnet can accept either an IP address or a domain name as the remote system address.

The Telnet server is enabled by default on the Cisco Nexus device.

Guidelines and Limitations for SSH

SSH has the following configuration guidelines and limitations:

- The Cisco Nexus device supports only SSH version 2 (SSHv2).

Configuring SSH

Generating SSH Server Keys

You can generate an SSH server key based on your security requirements. The default SSH server key is an RSA key that is generated using 1024 bits.

Procedure

	Command or Action	Purpose
Step 1	<code>switch# configure terminal</code>	Enters configuration mode.
Step 2	<code>switch(config)# ssh key {dsa [force] rsa [bits] [force]}</code>	Generates the SSH server key. The <i>bits</i> argument is the number of bits used to generate the key. The range is from 768 to 2048 and the default value is 1024.

	Command or Action	Purpose
		Use the force keyword to replace an existing key.
Step 3	switch(config)# exit	Exits global configuration mode.
Step 4	(Optional) switch# show ssh key	Displays the SSH server keys.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to generate an SSH server key:

```
switch# configure terminal
switch(config)# ssh key rsa 2048
switch(config)# exit
switch# show ssh key
switch# copy running-config startup-config
```

Specifying the SSH Public Keys for User Accounts

You can configure an SSH public key to log in using an SSH client without being prompted for a password. You can specify the SSH public key in one of three different formats:

- Open SSH format
- IETF SECSH format
- Public Key Certificate in PEM format

Specifying the SSH Public Keys in Open SSH Format

You can specify the SSH public keys in SSH format for user accounts.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# username <i>username</i> sshkey <i>ssh-key</i>	Configures the SSH public key in SSH format.
Step 3	switch(config)# exit	Exits global configuration mode.
Step 4	(Optional) switch# show user-account	Displays the user account configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to specify an SSH public key in open SSH format:

```
switch# configure terminal

switch(config)# username User1 sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAri3mQy4W1AV9Y2t2hrEWgbUEYz
CfTPO5B8LRkedn56BEy2N9ZcdpqE6aqJLZwfZcTFEzaAAZp9AS86dgBAjsKGS7UxnhGySr8ZELv+DQBsDQH6rZt0KR+2Da8hJD4Z
XIeccWk0gS1DQUNZ300xstQsYZUtqnx1bvm5Ninn0McNinn0Mc=

switch(config)# exit

switch# show user-account

switch# copy running-config startup-config
```



Note The **username** command in the example above is a single line that has been broken for legibility.

Specifying the SSH Public Keys in IETF SECSH Format

You can specify the SSH public keys in IETF SECSH format for user accounts.

Procedure

	Command or Action	Purpose
Step 1	switch# copy <i>server-file</i> bootflash: <i>filename</i>	Downloads the file that contains the SSH key in IETF SECSH format from a server. The server can be FTP, SCP, SFTP, or TFTP.
Step 2	switch# configure terminal	Enters configuration mode.
Step 3	switch(config)# username <i>username</i> sshkey file <i>filename</i>	Configures the SSH public key in SSH format.
Step 4	switch(config)# exit	Exits global configuration mode.
Step 5	(Optional) switch# show user-account	Displays the user account configuration.
Step 6	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to specify the SSH public key in the IETF SECSH format:

```
switch# copy tftp://10.10.1.1/secsh_file.pub bootflash:secsh_file.pub

switch# configure terminal

switch(config)# username User1 sshkey file bootflash:secsh_file.pub
```



```
switch(config)# exit
switch# show user-account
switch# copy running-config startup-config
```

Specifying the SSH Public Keys in PEM-Formatted Public Key Certificate Form

You can specify the SSH public keys in PEM-formatted Public Key Certificate form for user accounts.

Procedure

	Command or Action	Purpose
Step 1	switch# copy <i>server-file</i> bootflash: <i>filename</i>	Downloads the file that contains the SSH key in PEM-formatted Public Key Certificate form from a server. The server can be FTP, SCP, SFTP, or TFTP
Step 2	switch# configure terminal	Enters configuration mode.
Step 3	(Optional) switch# show user-account	Displays the user account configuration.
Step 4	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to specify the SSH public keys in PEM-formatted public key certificate form:

```
switch# copy tftp://10.10.1.1/cert.pem bootflash:cert.pem
switch# configure terminal
switch# show user-account
switch# copy running-config startup-config
```

Starting SSH Sessions to Remote Devices

You can start SSH sessions to connect to remote devices from your Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# ssh { <i>hostname</i> <i>username@hostname</i> } [vrf <i>vrf-name</i>]	Creates an SSH session to a remote device. The <i>hostname</i> argument can be an IPv4 address, an IPv6 address, or a hostname.

Clearing SSH Hosts

When you download a file from a server using SCP or SFTP, you establish a trusted SSH relationship with that server.

Procedure

	Command or Action	Purpose
Step 1	switch# clear ssh hosts	Clears the SSH host sessions.

Disabling the SSH Server

By default, the SSH server is enabled on the Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# no feature ssh	Disables the SSH server. The default is enabled.
Step 3	switch(config)# exit	Exits global configuration mode.
Step 4	(Optional) switch# show ssh server	Displays the SSH server configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Deleting SSH Server Keys

You can delete SSH server keys after you disable the SSH server.

**Note**

To reenable SSH, you must first generate an SSH server key.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# no feature ssh	Disables the SSH server.
Step 3	switch(config)# no ssh key [dsa rsa]	Deletes the SSH server key. The default is to delete all the SSH keys.
Step 4	switch(config)# exit	Exits global configuration mode.
Step 5	(Optional) switch# show ssh key	Displays the SSH server configuration.
Step 6	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Clearing SSH Sessions

You can clear SSH sessions from the Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# show users	Displays user session information.
Step 2	switch# clear line vty-line	Clears a user SSH session.

Configuration Examples for SSH

The following example shows how to configure SSH:

Procedure

Step 1 Generate an SSH server key.

```
switch(config)# ssh key rsa
generating rsa key(1024 bits).....
.
generated rsa key
```

Step 2 Enable the SSH server.

```
switch# configure terminal
switch(config)# feature ssh
```

Note This step should not be required because the SSH server is enabled by default.

Step 3 Display the SSH server key.

```
switch(config)# show ssh key
rsa Keys generated:Fri May 8 22:09:47 2009

ssh-rsa
AAAAB3NzaClyc2EAAAABIwAAAIEAri3mQy4W1AV9Y2t2hrEWgbUEYzCfTPO5B8LRkedn56BEy2N9ZcdpqE6aqJLZwfZ/
cTFEzaAAZp9AS86dgBAjsKGS7UxnhGySr8ZELv+DQBsDQH6rZt0KR+2Da8hJD4ZXIeccWk0gS1DQUNZ300xstQsYZUtqnx1bvm5/
Ninn0Mc=

bitcount:1024
fingerprint:
4b:4d:f6:b9:42:e9:d9:71:3c:bd:09:94:4a:93:ac:ca
*****
could not retrieve dsa key information
```

```
*****
```

Step 4 Specify the SSH public key in Open SSH format.

```
switch(config)# username User1 sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAri3mQy4W1AV9Y2t2hrEWgbUEYz
CftPO5B8LRkedn56BEy2N9ZcdpqE6aqJLZwfZcTFEzaAAZp9AS86dgBAjsKGS7UxnhGySr8ZELv+DQBsDQH6rZt0KR+2Da8hJD4Z
XIeccWk0gS1DQUNZ300xstQsYZUtqnx1bvm5Ninn0McNinn0Mc=
```

Step 5 Save the configuration.

```
switch(config)# copy running-config startup-config
```

Configuring Telnet

Enabling the Telnet Server

By default, the Telnet server is enabled. You can disable the Telnet server on your Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# feature telnet	Disables the Telnet server. The default is enabled.

Reenabling the Telnet Server

If the Telnet server on your Cisco Nexus device has been disabled, you can reenabling it.

Procedure

	Command or Action	Purpose
Step 1	switch(config)# feature telnet	Reenables the Telnet server.

Starting Telnet Sessions to Remote Devices

Before you start a Telnet session to connect to remote devices, you should do the following:

- Obtain the hostname for the remote device and, if needed, obtain the username on the remote device.
- Enable the Telnet server on the Cisco Nexus device.
- Enable the Telnet server on the remote device.

Procedure

	Command or Action	Purpose
Step 1	switch# telnet <i>hostname</i>	Creates a Telnet session to a remote device. The <i>hostname</i> argument can be an IPv4 address, an IPv6 address, or a device name.

Example

The following example shows how to start a Telnet session to connect to a remote device:

```
switch# telnet 10.10.1.1
Trying 10.10.1.1...
Connected to 10.10.1.1.
Escape character is '^]'.
switch login:
```

Clearing Telnet Sessions

You can clear Telnet sessions from the Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# show users	Displays user session information.
Step 2	switch# clear line <i>vtty-line</i>	Clears a user Telnet session.

Verifying the SSH and Telnet Configuration

To display the SSH configuration information, perform one of the following tasks:

Procedure

- switch# **show ssh key** [**dsa** | **rsa**]
Displays SSH server key-pair information.
- switch# **show running-config security** [**all**]
Displays the SSH and user account configuration in the running configuration. The **all** keyword displays the default values for the SSH and user accounts.
- switch# **show ssh server**
Displays the SSH server configuration.
- switch# **show user-account**
Displays user account information.

Default Settings for SSH

The following table lists the default settings for SSH parameters.

Table 8: Default SSH Parameters

Parameters	Default
SSH server	Enabled
SSH server key	RSA key generated with 1024 bits
RSA key bits for generation	1024
Telnet server	Enabled



CHAPTER 7

Configuring 802.1X

This chapter contains the following sections:

- [Information About 802.1X, on page 69](#)
- [Licensing Requirements for 802.1X, on page 75](#)
- [Prerequisites for 802.1X, on page 76](#)
- [802.1X Guidelines and Limitations, on page 76](#)
- [Default Settings for 802.1X, on page 77](#)
- [Configuring 802.1X, on page 78](#)
- [Verifying the 802.1X Configuration, on page 96](#)
- [Monitoring 802.1X, on page 96](#)
- [Configuration Example for 802.1X, on page 96](#)
- [Additional References for 802.1X, on page 97](#)
- [Feature History for 802.1X, on page 98](#)

Information About 802.1X

802.1X defines a client-server-based access control and authentication protocol that restricts unauthorized clients from connecting to a LAN through publicly accessible ports. The authentication server authenticates each client connected to a Cisco NX-OS device port.

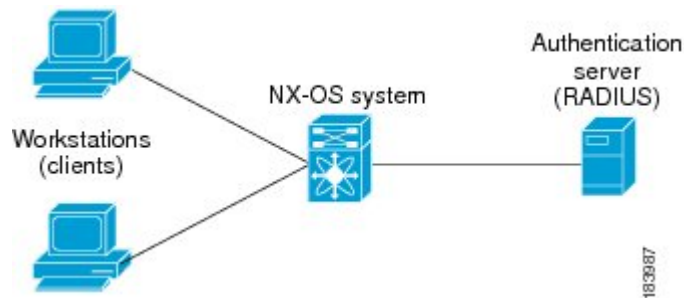
Until the client is authenticated, 802.1X access control allows only Extensible Authentication Protocol over LAN (EAPOL) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

Device Roles

With 802.1X port-based authentication, the devices in the network have specific roles.

Figure 4: 802.1X Device Roles

This figure shows the device roles in 802.1X.



The specific roles are as follows:

Supplicant

The client device that requests access to the LAN and Cisco NX-OS device services and responds to requests from the Cisco NX-OS device. The workstation must be running 802.1X-compliant client software such as that offered in the Microsoft Windows XP operating device.



Note To resolve Windows XP network connectivity and Cisco 802.1X port-based authentication issues, read the Microsoft Knowledge Base article at this URL:
<http://support.microsoft.com/support/kb/articles/Q303/5/97.ASP>

Authentication server

The authentication server performs the actual authentication of the supplicant. The authentication server validates the identity of the supplicant and notifies the Cisco NX-OS device regarding whether the supplicant is authorized to access the LAN and Cisco NX-OS device services. Because the Cisco NX-OS device acts as the proxy, the authentication service is transparent to the supplicant. The Remote Authentication Dial-In User Service (RADIUS) security device with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server; it is available in Cisco Secure Access Control Server, version 3.0. RADIUS uses a supplicant-server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.

Authenticator

The authenticator controls the physical access to the network based on the authentication status of the supplicant. The authenticator acts as an intermediary (proxy) between the supplicant and the authentication server, requesting identity information from the supplicant, verifying the requested identity information with the authentication server, and relaying a response to the supplicant. The authenticator includes the RADIUS client, which is responsible for encapsulating and decapsulating the EAP frames and interacting with the authentication server.

When the authenticator receives EAPOL frames and relays them to the authentication server, the authenticator strips off the Ethernet header and encapsulates the remaining EAP frame in the RADIUS format. This encapsulation process does not modify or examine the EAP frames, and the authentication server must support EAP within the native frame format. When the authenticator receives frames from the authentication server, the authenticator removes the server's frame header, leaving the EAP frame, which the authenticator then encapsulates for Ethernet and sends to the supplicant.



Note The Cisco NX-OS device can only be an 802.1X authenticator.

Authentication Initiation and Message Exchange

Either the authenticator (Cisco NX-OS device) or the supplicant (client) can initiate authentication. If you enable authentication on a port, the authenticator must initiate authentication when it determines that the port link state transitions from down to up. The authenticator then sends an EAP-request/identity frame to the supplicant to request its identity (typically, the authenticator sends an initial identity/request frame followed by one or more requests for authentication information). When the supplicant receives the frame, it responds with an EAP-response/identity frame.

If the supplicant does not receive an EAP-request/identity frame from the authenticator during bootup, the supplicant can initiate authentication by sending an EAPOL-start frame, which prompts the authenticator to request the supplicant's identity.



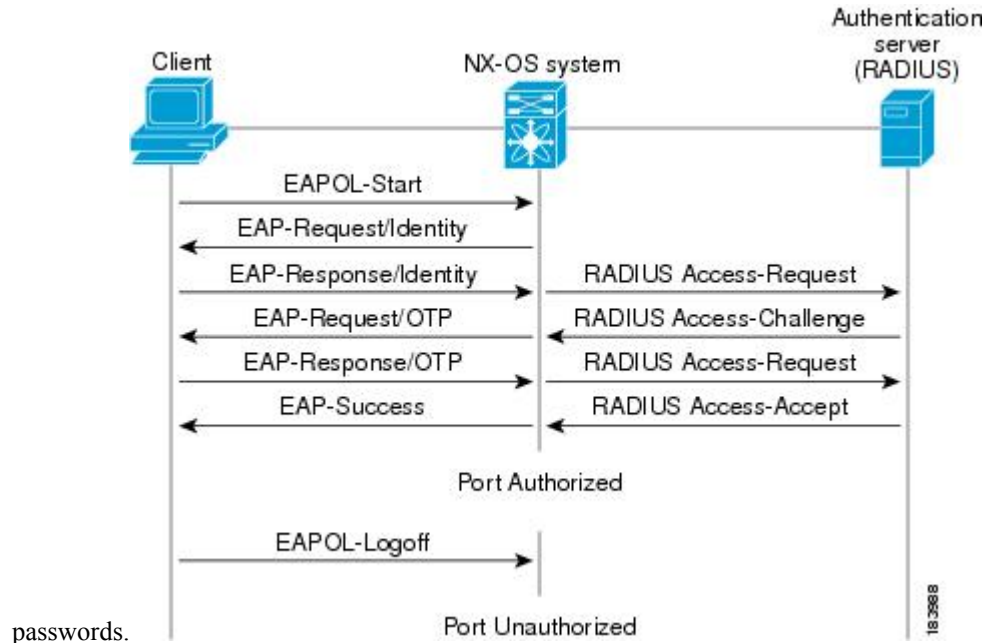
Note If 802.1X is not enabled or supported on the network access device, the Cisco NX-OS device drops any EAPOL frames from the supplicant. If the supplicant does not receive an EAP-request/identity frame after three attempts to start authentication, the supplicant transmits data as if the port is in the authorized state. A port in the authorized state means that the supplicant has been successfully authenticated.

When the supplicant supplies its identity, the authenticator begins its role as the intermediary, passing EAP frames between the supplicant and the authentication server until authentication succeeds or fails. If the authentication succeeds, the authenticator port becomes authorized.

The specific exchange of EAP frames depends on the authentication method being used.

Figure 5: Message Exchange

This figure shows a message exchange initiated by the supplicant using the One-Time-Password (OTP) authentication method with a RADIUS server. The OTP authentication device uses a secret pass-phrase to generate a sequence of one-time (single use)



The user's secret pass-phrase never crosses the network at any time such as during authentication or during pass-phrase changes.

Authenticator PAE Status for Interfaces

When you enable 802.1X on an interface, the Cisco NX-OS software creates an authenticator port access entity (PAE) instance. An authenticator PAE is a protocol entity that supports authentication on the interface. When you disable 802.1X on the interface, the Cisco NX-OS software does not automatically clear the authenticator PAE instances. You can explicitly remove the authenticator PAE from the interface and then reapply it, as needed.

Ports in Authorized and Unauthorized States

The authenticator port state determines if the supplicant is granted access to the network. The port starts in the unauthorized state. In this state, the port disallows all ingress and egress traffic except for 802.1X protocol packets. When a supplicant is successfully authenticated, the port transitions to the authorized state, allowing all traffic for the supplicant to flow normally.

If a client that does not support 802.1X is connected to an unauthorized 802.1X port, the authenticator 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 protocol, 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.

Ports can have the following authorization states:

Force authorized

Disables 802.1X port-based authentication and transitions to the authorized state without requiring any authentication exchange. The port transmits and receives normal traffic without 802.1X-based authentication of the client. This authorization state is the default.

Force unauthorized

Causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The authenticator cannot provide authentication services to the client through the interface.

Auto

Enables 802.1X port-based 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 transitions from down to up or when an EAPOL-start frame is received from the supplicant. The authenticator requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each supplicant that attempts to access the network is uniquely identified by the authenticator by using the supplicant's MAC address.

If the supplicant is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated supplicant 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 authenticator can retransmit the request. If no response is received from the server after the specified number of attempts, authentication fails, and the supplicant is not granted network access.

When a supplicant logs off, it sends an EAPOL-logoff message, which causes the authenticator port to transition to the unauthorized state.

If the link state of a port transitions from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

MAC Authentication Bypass

You can configure the Cisco NX-OS device to authorize a supplicant based on the supplicant MAC address by using the MAC authentication bypass feature. For example, you can enable this feature on interfaces configured for 802.1X that are connected to devices such as printers.

If 802.1X authentication times out while waiting for an EAPOL response from the supplicant, the Cisco NX-OS device tries to authorize the client by using MAC authentication bypass.

When you enable the MAC authentication bypass feature on an interface, the Cisco NX-OS device uses the MAC address as the supplicant identity. The authentication server has a database of supplicant MAC addresses that are allowed network access. After detecting a client on the interface, the Cisco NX-OS device waits for an Ethernet packet from the client. The Cisco NX-OS device sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the Cisco NX-OS device grants the client access to the network.

If an EAPOL packet is detected on the interface during the lifetime of the link, the Cisco NX-OS device 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 Cisco NX-OS device already authorized an interface by using MAC authentication bypass and detects an 802.1X supplicant, the Cisco NX-OS device does not unauthorize the client connected to the interface. When reauthentication occurs, the Cisco NX-OS device uses 802.1X authentication as the preferred reauthentication process if the previous session ended because the Termination-Action RADIUS attribute value is DEFAULT.

Clients that were authorized with MAC authentication bypass can be reauthenticated. The reauthentication process is the same as that for clients that were authenticated with 802.1X. During reauthentication, the port remains in the previously assigned VLAN. If reauthentication is successful, the switch keeps the port in the same VLAN.

If reauthentication 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 reauthentication. If MAC authentication bypass is enabled and the 802.1X authentication times out, the switch uses the MAC authentication bypass feature to initiate reauthorization. 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 following features:

- 802.1X authentication—You can enable MAC authentication bypass only if 802.1X authentication is enabled on the port.
- Port security— You can configure 802.1X authentication and port security on the same Layer 2 ports.
- Network admission control (NAC) Layer 2 IP validation—This feature takes effect after an 802.1X port is authenticated with MAC authentication bypass, including hosts in the exception list.

802.1X and Port Security

You can configure port security and 802.1X on the same interfaces. Port security secures the MAC addresses that 802.1X authenticates. 802.1X processes packets before port security processes them, so when you enable both on an interface, 802.1X is already preventing inbound traffic on the interface from unknown MAC addresses.

When you enable 802.1X and port security on the same interface, port security continues to learn MAC addresses by the sticky or dynamic method, as configured. Additionally, depending on whether you enable 802.1X in single-host mode or multiple-host mode, one of the following occurs:

Single host mode

Port security learns the MAC address of the authenticated host.

Multiple host mode

Port security drops any MAC addresses learned for this interface by the dynamic method and learns the MAC address of the first host authenticated by 802.1X.

If a MAC address that 802.1X passes to port security would violate the applicable maximum number of secure MAC addresses, the device sends an authentication failure message to the host.

The device treats MAC addresses authenticated by 802.1X as though they were learned by the dynamic method, even if port security previously learned the address by the sticky or static methods. If you attempt to delete a secure MAC address that has been authenticated by 802.1X, the address remains secure.

If the MAC address of an authenticated host is secured by the sticky or static method, the device treats the address as if it were learned by the dynamic method, and you cannot delete the MAC address manually.

Port security integrates with 802.1X to reauthenticate hosts when the authenticated and secure MAC address of the host reaches its port security age limit. The device behaves differently depending upon the type of aging, as follows:

Absolute

Port security notifies 802.1X and the device attempts to reauthenticate the host. The result of reauthentication determines whether the address remains secure. If reauthentication succeeds, the device restarts the aging timer on the secure address; otherwise, the device drops the address from the list of secure addressees for the interface.

Inactivity

Port security drops the secure address from the list of secure addresses for the interface and notifies 802.1X. The device attempts to reauthenticate the host. If reauthentication succeeds, port security secures the address again.

Single Host and Multiple Hosts Support

The 802.1X feature can restrict traffic on a port to only one endpoint device (single-host mode) or allow traffic from multiple endpoint devices on a port (multi-host mode).

Single-host mode allows traffic from only one endpoint device on the 802.1X port. Once the endpoint device is authenticated, the Cisco NX-OS device puts the port in the authorized state. When the endpoint device leaves the port, the Cisco NX-OS device put the port back into the unauthorized state. A security violation in 802.1X is defined as a detection of frames sourced from any MAC address other than the single MAC address authorized as a result of successful authentication. In this case, the interface on which this security association violation is detected (EAPOL frame from the other MAC address) will be disabled. Single host mode is

applicable only for host-to-switch topology and when a single host is connected to the Layer 2 (Ethernet access port) or Layer 3 port (routed port) of the Cisco NX-OS device.

Only the first host has to be authenticated on the 802.1X port configured with multiple host mode. The port is moved to the authorized state after the successful authorization of the first host. Subsequent hosts are not required to be authorized to gain network access once the port is in the authorized state. If the port becomes unauthorized when reauthentication fails or an EAPOL logoff message is received, all attached hosts are denied access to the network. The capability of the interface to shut down upon security association violation is disabled in multiple host mode. This mode is applicable for both switch-to-switch and host-to-switch topologies.

Supported Topologies

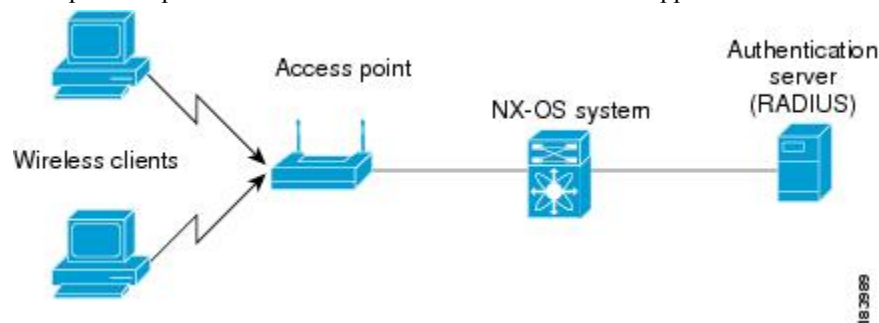
The 802.1X port-based authentication is supported in two topologies:

- Point-to-point
- Wireless LAN

In a point-to-point configuration, only one supplicant (client) can connect to the 802.1X-enabled authenticator (Cisco NX-OS device) port. The authenticator detects the supplicant when the port link state changes to the up state. If a supplicant leaves or is replaced with another supplicant, the authenticator changes the port link state to down, and the port returns to the unauthorized state.

Figure 6: Wireless LAN Example

This figure shows 802.1X port-based authentication in a wireless LAN. The 802.1X port is configured as a multiple-host port that becomes authorized as soon as one supplicant is authenticated.



When the port is authorized, all other hosts indirectly attached to the port are granted access to the network. If the port becomes unauthorized (reauthentication fails or an EAPOL-logoff message is received), the Cisco NX-OS device denies access to the network to all of the attached supplicants.

Licensing Requirements for 802.1X

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	802.1X requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For an explanation of the Cisco NX-OS licensing scheme, see the <i>Cisco NX-OS Licensing Guide</i> .

Prerequisites for 802.1X

802.1X has the following prerequisites:

- Cisco Nexus Release 6.0(2)N1(2) software.
- One or more RADIUS servers are accessible in the network.
- 802.1X supplicants are attached to the ports, unless you enable MAC address authentication bypass.

802.1X Guidelines and Limitations

802.1X port-based authentication has the following configuration guidelines and limitations:

- The Cisco NX-OS software supports 802.1X authentication only on physical ports.
- The Cisco NX-OS software does not support 802.1X authentication on port channels or subinterfaces.
- The Cisco NX-OS software supports 802.1X authentication on member ports of a port channel but not on the port channel itself.
- The Cisco NX-OS software does not support the following 802.1X configurations on port channel members when the members are configured for 802.1X:
 - Single-host mode cannot be configured on member ports of a port channel. Only multi-host mode is supported on member ports of a port channel.
 - MAC authentication bypass cannot be enabled on the member ports.
 - Port security cannot be configured on the port channel.
- Member ports with and without 802.1X configuration can coexist in a port channel. However, you must ensure the identical 802.1X configuration on all the member ports in order for channeling to operate with 802.1X.
- When you enable 802.1X authentication, supplicants are authenticated before any other Layer 2 or Layer 3 features are enabled on an Ethernet interface.
- The Cisco NX-OS software supports 802.1X authentication only on Ethernet interfaces that are in a port channel or a trunk or an access port.
- The Cisco NX-OS software does not support single host mode on trunk interfaces or member interfaces in a port channel.
- The Cisco NX-OS software does not support MAC address authentication bypass on trunk interfaces.
- The Cisco NX-OS software does not support MAC address authentication bypass on a port channel.
- The Cisco NX-OS software does not support Dot1X on vPC ports and MCT.
- The Cisco NX-OS software does not support the following 802.1X protocol enhancements:
 - One-to-many logical VLAN name to ID mapping
 - Web authorization

- Dynamic domain bridge assignment
- IP telephony
- Guest VLANs

Default Settings for 802.1X

This table lists the default settings for 802.1X parameters.

Table 9: Default 802.1X Parameters

Parameters	Default
802.1X feature	Disabled
AAA 802.1X authentication method	Not configured
Per-interface 802.1X protocol enable state	Disabled (force-authorized) Note The port transmits and receives normal traffic without 802.1X-based authentication of the supplicant.
Periodic reauthentication	Disabled
Number of seconds between reauthentication attempts	3600 seconds
Quiet timeout period	60 seconds (number of seconds that the Cisco NX-OS device remains in the quiet state following a failed authentication exchange with the supplicant)
Retransmission timeout period	30 seconds (number of seconds that the Cisco NX-OS device should wait for a response to an EAP request/identity frame from the supplicant before retransmitting the request)
Maximum retransmission number	2 times (number of times that the Cisco NX-OS device will send an EAP-request/identity frame before restarting the authentication process)
Host mode	Single host
Supplicant timeout period	30 seconds (when relaying a request from the authentication server to the supplicant, the amount of time that the Cisco NX-OS device waits for a response before retransmitting the request to the supplicant)
Authentication server timeout period	30 seconds (when relaying a response from the supplicant to the authentication server, the amount of time that the Cisco NX-OS device waits for a reply before retransmitting the response to the server)

Configuring 802.1X

This section describes how to configure the 802.1X feature.



Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Process for Configuring 802.1X

This section describes the process for configuring 802.1X.

Procedure

- Step 1** Enable the 802.1X feature.
- Step 2** Configure the connection to the remote RADIUS server.
- Step 3** Enable 802.1X feature on the Ethernet interfaces.

Enabling the 802.1X Feature

You must enable the 802.1X feature on the Cisco NX-OS device before authenticating any supplicant devices.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	feature dot1x Example: <pre>switch(config)# feature dot1x</pre>	Enables the 802.1X feature. The default is disabled.
Step 3	exit Example: <pre>switch(config)# exit switch#</pre>	Exits configuration mode.
Step 4	(Optional) show dot1x Example: <pre>switch# show dot1x</pre>	Displays the 802.1X feature status.

	Command or Action	Purpose
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring AAA Authentication Methods for 802.1X

You can use remote RADIUS servers for 802.1X authentication. You must configure RADIUS servers and RADIUS server groups and specify the default AAA authentication method before the Cisco NX-OS device can perform 802.1X authentication.

Before you begin

Obtain the names or addresses for the remote RADIUS server groups.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	aaa authentication dot1x default group group-list Example: switch(config)# aaa authentication dot1x default group rad2	Specifies the RADIUS server groups to use for 802.1X authentication. The <i>group-list</i> argument consists of a space-delimited list of group names. The group names are the following: <ul style="list-style-type: none"> • radius—Uses the global pool of RADIUS servers for authentication. • named-group—Uses the global pool of RADIUS servers for authentication.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) show radius-server Example: switch# show radius-server	Displays the RADIUS server configuration.

	Command or Action	Purpose
Step 5	(Optional) show radius-server group [<i>group-name</i>] Example: switch# show radius-server group rad2	Displays the RADIUS server group configuration.
Step 6	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Controlling 802.1X Authentication on an Interface

You can control the 802.1X authentication performed on an interface. An interface can have the following 802.1X authentication states:

Auto

Enables 802.1X authentication on the interface.

Force-authorized

Disables 802.1X authentication on the interface and allows all traffic on the interface without authentication. This state is the default.

Force-unauthorized

Disallows all traffic on the interface.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet slot / port Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Selects the interface to configure and enters interface configuration mode.
Step 3	dot1x port-control {auto force-authorized forced-unauthorized} Example: switch(config-if)# dot1x port-control auto	Changes the 802.1X authentication state on the interface. The default is force-authorized.

	Command or Action	Purpose
Step 4	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 5	(Optional) show dot1x all Example: switch# show dot1x all	Displays all 802.1X feature status and configuration information.
Step 6	(Optional) show dot1x interface ethernet slot / port Example: switch# show dot1x interface ethernet 2/1	Displays 802.1X feature status and configuration information for an interface.
Step 7	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring 802.1X Authentication on Member Ports

You can configure 802.1X authentication on the members of a port channel.



Note You cannot configure 802.1X authentication on the port channel itself.

There are two ways to configure 802.1X authentication on member ports: 1) by configuring 802.1X on a member port and then adding the port to a port channel or 2) by creating a port channel, adding a port to the port channel, and then configuring 802.1X on the port. The following procedure provides instructions for the first method. To configure 802.1X using the second method, use these commands:

- **interface port-channel** *channel-number*
- **interface ethernet** *slot/port*
- **channel-group** *channel-number* [**force**] [**mode** {**on** | **active** | **passive**}]
- **dot1x port-control** **auto**



Note For more information on the above commands, see the *Cisco NX-OS Interfaces Command Reference* for your platform.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	<p>interface ethernet <i>slot/port</i></p> <p>Example:</p> <pre>switch(config)# interface ethernet 7/1 switch(config-if)#</pre>	Selects the interface to configure and enters interface configuration mode.
Step 3	<p>dot1x port-control auto</p> <p>Example:</p> <pre>switch(config-if)# dot1x port-control auto</pre>	Changes the 802.1X authentication state on the interface.
Step 4	<p>[no] switchport</p> <p>Example:</p> <pre>switch(config-if)# switchport</pre>	Configures the interface as a Layer 2 port or, if you use the no keyword, as a Layer 3 port.
Step 5	<p>dot1x host-mode multi-host</p> <p>Example:</p> <pre>switch(config-if)# dot1x host-mode multi-host</pre>	Enables multiple hosts mode for the interface. This command is required in order to add a port to a port channel.
Step 6	<p>channel-group <i>channel-number</i> [force] [mode {on active passive}]</p> <p>Example:</p> <pre>switch(config-if)# channel-group 5 force</pre>	<p>Configures the port in a channel group and sets the mode. The channel number range is from 1 to 4096. The Cisco NX-OS software creates the port channel associated with this channel group if the port channel does not already exist.</p> <p>The optional force keyword allows you to force an interface with some incompatible configurations to join the channel. The forced interface must have the same speed, duplex, and flow control settings as the channel group.</p> <p>Note To remove an 802.1X-enabled port from a port channel, use the no channel-group <i>channel-number</i> command.</p>
Step 7	<p>exit</p> <p>Example:</p>	Exits interface configuration mode.

	Command or Action	Purpose
	<code>switch(config-if)# exit</code> <code>switch(config)#</code>	
Step 8	exit Example: <code>switch(config)# exit</code> <code>switch#</code>	Exits global configuration mode.
Step 9	(Optional) show dot1x all Example: <code>switch# show dot1x all</code>	Displays all 802.1X feature status and configuration information.
Step 10	(Optional) show dot1x interface ethernet slot/port Example: <code>switch# show dot1x interface ethernet 7/1</code>	Displays 802.1X feature status and configuration information for an interface.
Step 11	(Optional) copy running-config startup-config Example: <code>switch# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Creating or Removing an Authenticator PAE on an Interface

You can create or remove the 802.1X authenticator port access entity (PAE) instance on an interface.



Note By default, the Cisco NX-OS software creates the authenticator PAE instance on the interface when you enable 802.1X on an interface.

Before you begin

Enable the 802.1X feature.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	(Optional) show dot1x interface ethernet slot/port Example: switch# show dot1x interface ethernet 2/1	Displays the 802.1X configuration on the interface.
Step 3	interface ethernet slot/port Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Selects the interface to configure and enters interface configuration mode.
Step 4	[no] dot1x pae authenticator Example: switch(config-if)# dot1x pae authenticator	Creates an authenticator PAE instance on the interface. Use the no form to remove the PAE instance from the interface. Note If an authenticator PAE already exists on the interface the dot1x pae authentication command does not change the configuration on the interface.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling Periodic Reauthentication for an Interface

You can enable periodic 802.1X reauthentication on an interface and specify how often it occurs. If you do not specify a time period before enabling reauthentication, the number of seconds between reauthentication defaults to the global value.



Note During the reauthentication process, the status of an already authenticated supplicant is not disrupted.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	interface ethernet <i>slot/port</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Selects the interface to configure and enters interface configuration mode.
Step 3	dot1x re-authentication Example: switch(config-if)# dot1x re-authentication	Enables periodic reauthentication of the supplicants connected to the interface. By default, periodic authentication is disabled.
Step 4	(Optional) dot1x timeout re-authperiod <i>seconds</i> Example: switch(config-if)# dot1x timeout re-authperiod 3300	Sets the number of seconds between reauthentication attempts. The default is 3600 seconds. The range is from 1 to 65535. Note This command affects the behavior of the Cisco NX-OS device only if you enable periodic reauthentication on the interface.
Step 5	exit Example: switch(config-if)# exit switch(config)#	Exits configuration mode.
Step 6	(Optional) show dot1x all Example: switch(config)# show dot1x all	Displays all 802.1X feature status and configuration information.
Step 7	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Manually Reauthenticating Supplicants

You can manually reauthenticate the supplicants for the entire Cisco NX-OS device or for an interface.



Note During the reauthentication process, the status of an already authenticated supplicant is not disrupted.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	dot1x re-authenticate [<i>interface slot/port</i>] Example: <pre>switch# dot1x re-authenticate interface 2/1</pre>	Reauthenticates the supplicants on the Cisco NX-OS device or on an interface.

Manually Initializing 802.1X Authentication

You can manually initialize the authentication for all supplicants on a Cisco NX-OS device or for a specific interface.



Note Initializing the authentication clears any existing authentication status before starting the authentication process for the client.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	dot1x initialize [<i>interface ethernet slot/port</i>] Example: <pre>switch# dot1x initialize interface ethernet 2/1</pre>	Initializes 802.1X authentication on the Cisco NX-OS device or on a specified interface.

Changing 802.1X Authentication Timers for an Interface

You can change the following 802.1X authentication timers on the Cisco NX-OS device interfaces:

Quiet-period timer

When the Cisco NX-OS device cannot authenticate the supplicant, the switch remains idle for a set period of time and then tries again. The quiet-period timer value determines the idle period. An authentication failure might occur because the supplicant provided an invalid password. You can provide a faster response time to the user by entering a smaller number than the default. The default is the value of the global quiet period timer. The range is from 1 to 65535 seconds.

Rate-limit timer

The rate-limit period throttles EAPOL-Start packets from supplicants that are sending too many EAPOL-Start packets. The authenticator ignores EAPOL-Start packets from supplicants that have successfully authenticated for the rate-limit period duration. The default value is 0 seconds and the authenticator processes all EAPOL-Start packets. The range is from 1 to 65535 seconds.

Switch-to-authentication-server retransmission timer for Layer 4 packets

The authentication server notifies the switch each time that it receives a Layer 4 packet. If the switch does not receive a notification after sending a packet, the Cisco NX-OS device waits a set period of time and then retransmits the packet. The default is 30 seconds. The range is from 1 to 65535 seconds.

Switch-to-supplicant retransmission timer for EAP response frames

The supplicant responds to the EAP-request/identity frame from the Cisco NX-OS device with an EAP-response/identity frame. If the Cisco NX-OS device does not receive this response, it waits a set period of time (known as the retransmission time) and then retransmits the frame. The default is 30 seconds. The range is from 1 to 65535 seconds.

Switch-to-supplicant retransmission timer for EAP request frames

The supplicant notifies the Cisco NX-OS device it that received the EAP request frame. If the authenticator does not receive this notification, it waits a set period of time and then retransmits the frame. The default is the value of the global retransmission period timer. The range is from 1 to 65535 seconds.



Note You should change the default values only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain supplicants and authentication servers.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet <i>slot/port</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)	Selects the interface to configure and enters interface configuration mode.
Step 3	(Optional) dot1x timeout quiet-period <i>seconds</i> Example: switch(config-if)# dot1x timeout quiet-period 25	Sets the number of seconds that the authenticator waits for a response to an EAP-request/identity frame from the supplicant before retransmitting the request. The default is the global number of seconds set for all interfaces. The range is from 1 to 65535 seconds.
Step 4	(Optional) dot1x timeout ratelimit-period <i>seconds</i> Example: switch(config-if)# dot1x timeout ratelimit-period 10	Sets the number of seconds that the authenticator ignores EAPOL-Start packets from supplicants that have successfully authenticated. The default value is 0 seconds. The range is from 1 to 65535 seconds.

	Command or Action	Purpose
Step 5	(Optional) dot1x timeout server-timeout <i>seconds</i> Example: <pre>switch(config-if)# dot1x timeout server-timeout 60</pre>	Sets the number of seconds that the Cisco NX-OS device waits before retransmitting a packet to the authentication server. The default is 30 seconds. The range is from 1 to 65535 seconds.
Step 6	(Optional) dot1x timeout supp-timeout <i>seconds</i> Example: <pre>switch(config-if)# dot1x timeout supp-timeout 20</pre>	Sets the number of seconds that the Cisco NX-OS device waits for the supplicant to respond to an EAP request frame before the Cisco NX-OS device retransmits the frame. The default is 30 seconds. The range is from 1 to 65535 seconds.
Step 7	(Optional) dot1x timeout tx-period <i>seconds</i> Example: <pre>switch(config-if)# dot1x timeout tx-period 40</pre>	Sets the number of seconds between the retransmission of EAP request frames when the supplicant does not send notification that it received the request. The default is the global number of seconds set for all interfaces. The range is from 1 to 65535 seconds.
Step 8	exit Example: <pre>switch(config)# exit switch#</pre>	Exits configuration mode.
Step 9	(Optional) show dot1x all Example: <pre>switch# show dot1x all</pre>	Displays the 802.1X configuration.
Step 10	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Enabling Single Host or Multiple Hosts Mode

You can enable single host or multiple hosts mode on an interface.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet <i>slot/port</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)	Selects the interface to configure and enters interface configuration mode.
Step 3	dot1x host-mode {multi-host single-host} Example: switch(config-if)# dot1x host-mode multi-host	Configures the host mode. The default is single-host. Note Make sure that the dot1x port-control interface configuration command is set to auto for the specified interface.
Step 4	exit Example: switch(config-if)# exit switch(config)#	Exits configuration mode.
Step 5	(Optional) show dot1x all Example: switch# show dot1x all	Displays all 802.1X feature status and configuration information.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling MAC Authentication Bypass

You can enable MAC authentication bypass on an interface that has no supplicant connected.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	interface ethernet <i>slot/port</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)	Selects the interface to configure and enters interface configuration mode.
Step 3	dot1x mac-auth-bypass [cap] Example: switch(config-if)# dot1x mac-auth-bypass	Enables MAC authentication bypass. The default is bypass disabled. Use the cap keyword to configure the Cisco NX-OS device to use EAP for authorization.
Step 4	exit Example: switch(config-if)# exit switch(config)#	Exits configuration mode.
Step 5	(Optional) show dot1x all Example: switch# show dot1x all	Displays all 802.1X feature status and configuration information.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Disabling 802.1X Authentication on the Cisco NX-OS Device

You can disable 802.1X authentication on the Cisco NX-OS device. By default, the Cisco NX-OS software enables 802.1X authentication after you enable the 802.1X feature. However, when you disable the 802.1X feature, the configuration is removed from the Cisco NX-OS device. The Cisco NX-OS software allows you to disable 802.1X authentication without losing the 802.1X configuration.



Note

When you disable 802.1X authentication, the port mode for all interfaces defaults to force-authorized regardless of the configured port mode. When you reenables 802.1X authentication, the Cisco NX-OS software restores the configured port mode on the interfaces.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	no dot1x system-auth-control Example: switch(config)# no dot1x system-auth-control	Disables 802.1X authentication on the Cisco NX-OS device. The default is enabled. Note Use the dot1x system-auth-control command to enable 802.1X authentication on the Cisco NX-OS device.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) show dot1x Example: switch# show dot1x	Displays the 802.1X feature status.
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Disabling the 802.1X Feature

You can disable the 802.1X feature on the Cisco NX-OS device.

When you disable 802.1X, all related configurations are automatically discarded. The Cisco NX-OS software creates an automatic checkpoint that you can use if you reenables 802.1X and want to recover the configuration. For more information, see the *Cisco NX-OS System Management Configuration Guide* for your platform.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	no feature dot1x Example: switch(config)# no feature dot1x	Disables 802.1X. Caution Disabling the 802.1X feature removes all 802.1X configuration.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Setting the Maximum Authenticator-to-Supplicant Frame Retransmission Retry Count for an Interface

You can set the maximum number of times that the Cisco NX-OS device retransmits authentication requests to the supplicant on an interface before the session times out. The default is 2 times and the range is from 1 to 10.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet <i>slot/port</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Selects the interface to configure and enters interface configuration mode.
Step 3	dot1x max-req <i>count</i> Example: switch(config-if)# dot1x max-req 3	Changes the maximum authorization request retry count. The default is 2 times and the range is from 1 to 10.

	Command or Action	Purpose
		Note Make sure that the dot1x port-control interface configuration command is set to auto for the specified interface.
Step 4	exit Example: switch(config)# exit switch#	Exits interface configuration mode.
Step 5	(Optional) show dot1x all Example: switch# show dot1x all	Displays all 802.1X feature status and configuration information.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling RADIUS Accounting for 802.1X Authentication

You can enable RADIUS accounting for the 802.1X authentication activity.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	dot1x radius-accounting Example: switch(config)# dot1x radius-accounting	Enables RADIUS accounting for 802.1X. The default is disabled.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.

	Command or Action	Purpose
Step 4	(Optional) show dot1x Example: switch# show dot1x	Displays the 802.1X configuration.
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring AAA Accounting Methods for 802.1X

You can enable AAA accounting methods for the 802.1X feature.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enters global configuration mode.
Step 2	aaa accounting dot1x default group <i>group-list</i>	Configures AAA accounting for 802.1X. The default is disabled. The <i>group-list</i> argument consists of a space-delimited list of group names. The group names are the following: <ul style="list-style-type: none"> • radius—For all configured RADIUS servers. • <i>named-group</i>—Any configured RADIUS server group name.
Step 3	exit	Exits configuration mode.
Step 4	(Optional) show aaa accounting	Displays the AAA accounting configuration.
Step 5	(Optional) copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to enable the 802.1x feature:

```
switch# configure terminal
switch(config)# aaa accounting dot1x default group radius
switch(config)# exit
```



```
switch# show aaa accounting
switch# copy running-config startup-config
```

Setting the Maximum Reauthentication Retry Count on an Interface

You can set the maximum number of times that the Cisco NX-OS device retransmits reauthentication requests to the supplicant on an interface before the session times out. The default is 2 times and the range is from 1 to 10.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet slot/port Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Selects the interface to configure and enters interface configuration mode.
Step 3	dot1x max-reauth-req retry-count Example: switch(config-if)# dot1x max-reauth-req 3	Changes the maximum reauthentication request retry count. The default is 2 times and the range is from 1 to 10.
Step 4	exit Example: switch(config)# exit switch#	Exits interface configuration mode.
Step 5	(Optional) show dot1x all Example: switch# show dot1x all	Displays all 802.1X feature status and configuration information.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying the 802.1X Configuration

To display 802.1X information, perform one of the following tasks:

Command	Purpose
<code>show dot1x</code>	Displays the 802.1X feature status.
<code>show dot1x all [details statistics summary]</code>	Displays all 802.1X feature status and configuration information.
<code>show dot1x interface ethernet slot/port [details statistics summary]</code>	Displays the 802.1X feature status and configuration information for an Ethernet interface.
<code>show running-config dot1x [all]</code>	Displays the 802.1X feature configuration in the running configuration.
<code>show startup-config dot1x</code>	Displays the 802.1X feature configuration in the startup configuration.

For detailed information about the fields in the output from these commands, see the *Cisco NX-OS Security Command Reference* for your platform.

Monitoring 802.1X

You can display the statistics that the Cisco NX-OS device maintains for the 802.1X activity.

Before you begin

Enable the 802.1X feature on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	<code>show dot1x {all interface ethernet slot/port} statistics</code> Example: <code>switch# show dot1x all statistics</code>	Displays the 802.1X statistics.

Configuration Example for 802.1X

The following example shows how to configure 802.1X for an access port:

```
feature dot1x
aaa authentication dot1x default group rad2
interface Ethernet2/1
dot1x pae-authenticator
```

```
dot1x port-control auto
```

The following example shows how to configure 802.1X for a trunk port:

```
feature dot1x
aaa authentication dot1x default group rad2
interface Ethernet2/1
dot1x pae-authenticator
dot1x port-control auto
dot1x host-mode multi-host
```



Note Repeat the **dot1x pae authenticator** and **dot1x port-control auto** commands for all interfaces that require 802.1X authentication.

Additional References for 802.1X

This section includes additional information related to implementing 802.1X.

Related Documents

Related Topic	Document Title
Cisco NX-OS Licensing	<i>Cisco NX-OS Licensing Guide</i>
Command reference	<i>Cisco Nexus 5000 Series NX-OS Security Command Reference</i>
VRF configuration	<i>Cisco Nexus 5000 Series NX-OS Unicast Routing Configuration guide</i>

Standards

Standards	Title
IEEE Std 802.1X- 2004 (Revision of IEEE Std 802.1X-2001)	<i>802.1X IEEE Standard for Local and Metropolitan Area Networks Port-Based Network Access Control</i>
RFC 2284	<i>PPP Extensible Authentication Protocol (EAP)</i>
RFC 3580	<i>IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines</i>

MIBs

MIBs	MIBs Link
• IEEE8021-PAE-MIB	To locate and download MIBs, go to the following URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Feature History for 802.1X

Table 10: Feature History for 802.1X

Feature Name	Release	Feature Information
802.1X	6.0(2)N1(2)	This feature was introduced.



CHAPTER 8

Configuring Cisco TrustSec

This chapter describes how to configure Cisco TrustSec on Cisco NX-OS devices.

This chapter includes the following sections:

- [Information About Cisco TrustSec](#) , on page 99
- [Licensing Requirements for Cisco TrustSec](#) , on page 104
- [Prerequisites for Cisco TrustSec](#) , on page 105
- [Guidelines and Limitations for Cisco TrustSec](#) , on page 105
- [Default Settings For Cisco TrustSec](#) , on page 106
- [Configuring Cisco TrustSec](#) , on page 106
- [Verifying the Cisco TrustSec Configuration](#), on page 129
- [Configuration Examples for Cisco TrustSec](#), on page 130
- [Additional References for Cisco TrustSec](#), on page 133
- [Feature History for Cisco TrustSec](#), on page 133

Information About Cisco TrustSec

This section provides information about Cisco TrustSec.

Cisco TrustSec Architecture

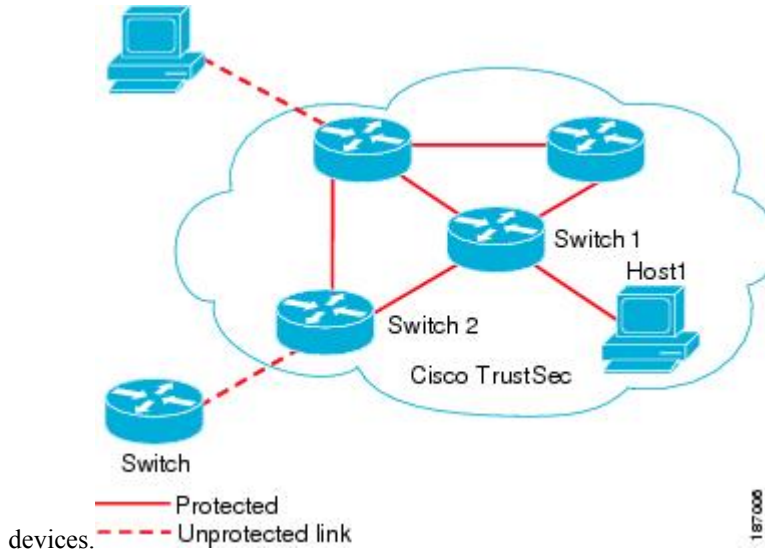
The Cisco TrustSec security architecture builds secure networks by establishing clouds of trusted network devices. Cisco TrustSec also uses the device information acquired during authentication for classifying, or coloring, the packets as they enter the network. This packet classification is maintained by tagging packets on ingress to the Cisco TrustSec network so that they can be properly identified for the purpose of applying security and other policy criteria along the data path. The tag, also called the security group tag (SGT), allows the network to enforce the access control policy by enabling the endpoint device to act upon the SGT to filter traffic.



Note Ingress refers to entering the first Cisco TrustSec-capable device encountered by a packet on its path to the destination and egress refers to leaving the last Cisco TrustSec-capable device on the path.

Figure 7: Cisco TrustSec Network Cloud Example

This figure shows an example of a Cisco TrustSec cloud. In this example, several networking devices and an endpoint device are inside the Cisco TrustSec cloud. One endpoint device and one networking device are outside the cloud because they are not Cisco TrustSec-capable.



The Cisco TrustSec architecture consists of the following major components:

Authentication

Verifies the identity of each device before allowing them to join the Cisco TrustSec network.

Authorization

Decides the level of access to the Cisco TrustSec network resources for a device based on the authenticated identity of the device.

Access control

Applies access policies on a per-packet basis using the source tags on each packet.

A Cisco TrustSec network has the following entities:

Authenticators (AT)

Devices that are already part of a Cisco TrustSec network.

Authorization server (AS)

Servers that may provide authentication information, authorization information, or both.

When the link first comes up, authorization occurs in which each side of the link obtains policies, such as SGT and ACLs, that apply to the link.

Authentication

Cisco TrustSec authenticates a device before allowing it to join the network.

Device Identities

Cisco TrustSec does not use IP addresses or MAC addresses as device identities. Instead, you assign a name (device ID) to each Cisco TrustSec-capable Cisco NX-OS device to identify it uniquely in the Cisco TrustSec network. This device ID used for the following:

- Looking up authorization policy

- Looking up passwords in the databases during authentication

Device Credentials

Cisco TrustSec supports password-based credentials. The authentication servers may use self-signed certificates instead. Cisco TrustSec authenticates the supplicants through passwords and uses MSCHAPv2 to provide mutual authentication even if the authentication server certificate is not verifiable.

The authentication server uses a temporarily configured password to authenticate the supplicant when the supplicant first joins the Cisco TrustSec network. When the supplicant first joins the Cisco TrustSec network, the authentication server authenticates the supplicant using a manufacturing certificate and then generates a strong password and pushes it to the supplicant with the PAC. The authentication server also keeps the new password in its database.

User Credentials

Cisco TrustSec does not require a specific type of user credentials for endpoint devices. You can choose any type of authentication method for the user (for example, MSCHAPv2, LEAP, generic token card (GTC), or OTP) and use the corresponding credentials.

SGACLs and SGTs

In security group access lists (SGACLs), you can control the operations that users can perform based on assigned security groups. The grouping of permissions into a role simplifies the management of the security policy. As you add users to the Cisco NX-OS device, you simply assign one or more security groups and they immediately receive the appropriate permissions. You can modify security groups to introduce new privileges or restrict current permissions.

Cisco TrustSec assigns a unique 16-bit tag, called the security group tag (SGT), to a security group. The number of SGTs in the Cisco NX-OS device is limited to the number of authenticated network entities. The SGT is a single label that indicates the privileges of the source within the entire enterprise. Its scope is global within a Cisco TrustSec network.

The management server derives the SGTs based on the security policy configuration. You do not have to configure them manually.

Once authenticated, Cisco TrustSec tags any packet that originates from a device with the SGT that represents the security group to which the device is assigned. The packet carries this SGT throughout the network within the Cisco TrustSec header. Because this tag represents the group of the source, the tag is referred to as the source SGT. At the egress edge of the network, Cisco TrustSec determines the group that is assigned to the packet destination device and applies the access control policy.

Cisco TrustSec defines access control policies between the security groups. By assigning devices within the network to security groups and applying access control between and within the security groups, Cisco TrustSec essentially achieves access control within the network.

Figure 8: SGACL Policy Example

This figure shows an example of an SGACL

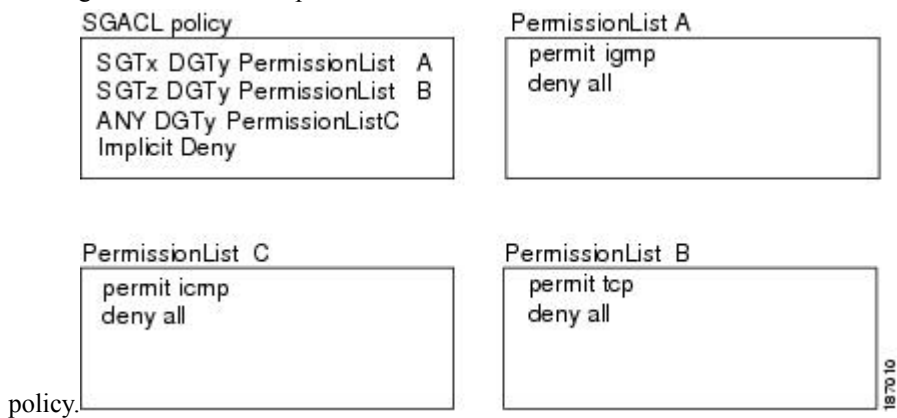
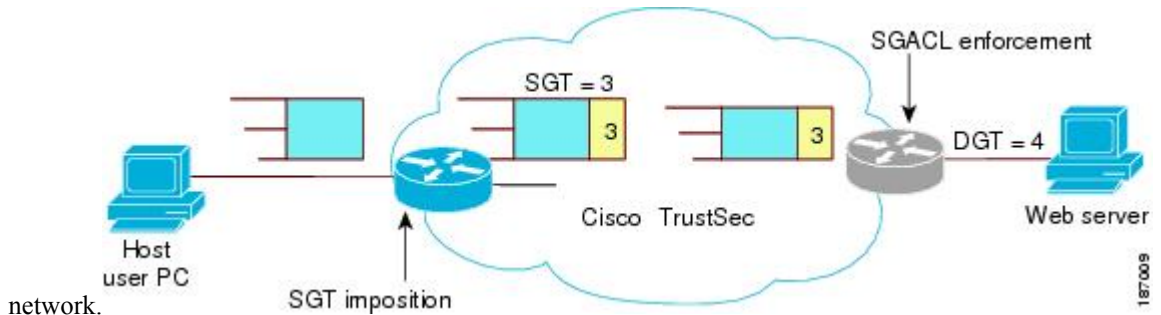


Figure 9: SGT and SGACL in Cisco TrustSec Network

This figure shows how the SGT assignment and the SGACL enforcement operate in a Cisco TrustSec



The Cisco NX-OS device defines Cisco TrustSec access control policy for a group of devices as opposed to IP addresses in traditional ACLs. With such a decoupling, the network devices are free to move throughout the network and change IP addresses. Entire network topologies can change. As long as the roles and the permissions remain the same, changes to the network do not change the security policy. This feature greatly reduces size of ACLs and simplifies their maintenance.

In traditional IP networks, the number of access control entries (ACEs) configured is determined as follows:

$$\# \text{ of ACEs} = (\# \text{ of sources specified}) \times (\# \text{ of destinations specified}) \times (\# \text{ of permissions specified})$$

Cisco TrustSec uses the following formula:

$$\# \text{ of ACEs} = \# \text{ of permissions specified}$$

Determining the Source Security Group

A network device at the ingress of the Cisco TrustSec cloud needs to determine the SGT of the packet entering the Cisco TrustSec cloud so that it can tag the packet with that SGT when it forwards it into the Cisco TrustSec cloud. The egress network device needs to determine the SGT of the packet so that it can apply the SGACLs.

The network device can determine the SGT for a packet in one of the following methods:

- Obtain the source SGT during policy acquisition—After the Cisco TrustSec authentication phase, a network device acquires a policy from an authentication server. Authentication server indicates whether

the peer device is trusted or not. If a peer device is not trusted then the authentication server can also provide an SGT to apply to all packets coming from the peer device.

- Obtain the source SGT field from the Cisco TrustSec header—If a packet comes from a trusted peer device, the Cisco TrustSec header carries the correct SGT field if the network device is not the first network device in the Cisco TrustSec cloud for the packet.

Determining the Destination Security Group

The egress network device in a Cisco TrustSec cloud determines the destination group for applying the SGACL. In some cases, ingress devices or other nonegress devices might have destination group information available. In those cases, SGACLs might be applied in these devices rather than in egress devices.

Cisco TrustSec determines the destination group for the packet based on the destination IP address.

You do not configure the destination SGT to enforce Cisco TrustSec on egress broadcast, multicast, and unknown unicast traffic on FEX or vEthernet ports. Instead, you set the DST to zero (unknown). The following is an example of the correct configuration:

```
cts role-based access-list acl-on-fex-egress
  deny udp
  deny ip
cts role-based sgt 9 dst 0 access-list acl-on-fex-egress
```

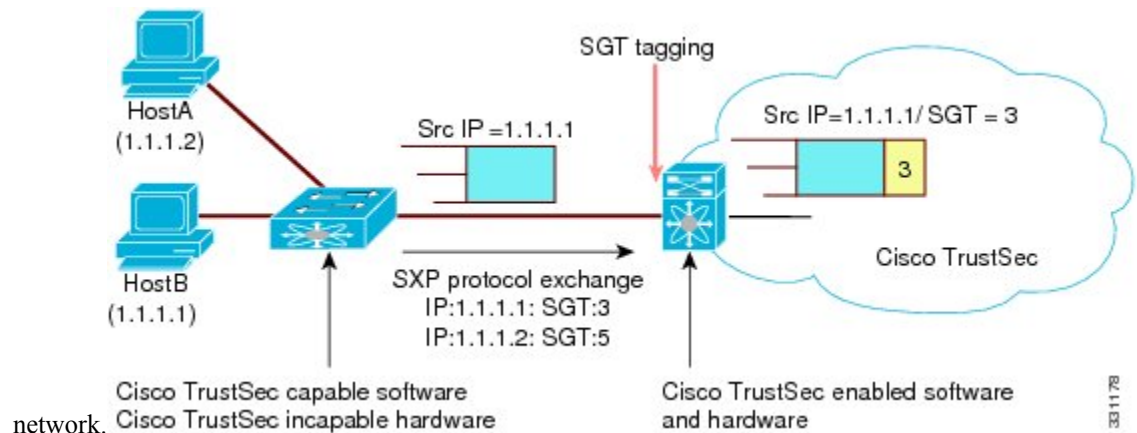
SXP for SGT Propagation Across Legacy Access Networks

The Cisco NX-OS device hardware in the access layer supports Cisco TrustSec. Without the Cisco TrustSec hardware, the Cisco TrustSec software cannot tag the packets with SGTs. You can use SXP to propagate the SGTs across network devices that do not have hardware support for Cisco TrustSec.

SXP operates between access layer devices and distribution layer devices. The access layer devices use SXP to pass the IP addresses of the Cisco TrustSec-authenticated devices with their SGTs to the distribution switches. Distribution devices with both Cisco TrustSec-enabled software and hardware can use this information to tag packets appropriately and enforce SGACL policies.

Figure 10: Using SXP to Propagate SGT Information

This figure shows how to use SXP to propagate SGT information in a legacy



Tagging packets with SGTs requires hardware support. You might have devices in your network that cannot tag packets with SGTs. To allow these devices to send IP address-to-SGT mappings to a device that has Cisco

TrustSec-capable hardware, you must manually set up the SXP connections. Manually setting up an SXP connection requires the following:

- If you require SXP data integrity and authentication, you must configure both the same SXP password on both of the peer devices. You can configure the SXP password either explicitly for each peer connection or globally for the device. The SXP password is not required.
- You must configure each peer on the SXP connection as either an SXP speaker or an SXP listener. The speaker device distributes the SXP information to the listener device.



Note This Cisco Nexus device does not have the functionality to be an SXP listener. It can only be an SXP speaker.

- You can specify a source IP address to use for each peer relationship or you can configure a default source IP address for peer connections where you have not configured a specific source IP address.

Environment Data Download

The Cisco TrustSec environment data is a collection of information or policies that assists a device to function as a Cisco TrustSec node. The device acquires the environment data from the authentication server when the device first joins a Cisco TrustSec cloud, although you might also manually configure some of the data on a device. For example, you must configure the seed Cisco TrustSec device with the authentication server information, which can later be augmented by the server list that the device acquires from the authentication server.

The device must refresh the Cisco TrustSec environment data before it expires. The device can also cache the data and reuse it after a reboot if the data has not expired.

The device uses RADIUS to acquire the following environment data from the authentication server:

Server lists

List of servers that the client can use for future RADIUS requests (for both authentication and authorization).

Device SGT

Security group to which the device itself belongs.

Expiry timeout

Interval that controls how often the Cisco TrustSec device should refresh its environment data.

Licensing Requirements for Cisco TrustSec

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	Cisco TrustSec requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the <i>License and Copyright Information for Cisco NX-OS Software</i> .

Prerequisites for Cisco TrustSec

Cisco TrustSec has the following prerequisites:

- You must enable the 802.1X feature before you enable the Cisco TrustSec feature. Although none of the 802.1X interface level features are available, 802.1X is required for the device to authenticate with RADIUS.
- You must enable the Cisco TrustSec feature.

Guidelines and Limitations for Cisco TrustSec

Cisco TrustSec has the following guidelines and limitations:

- Cisco TrustSec uses RADIUS for authentication.
- AAA authentication and authorization for Cisco TrustSec is only supported by the Cisco Secure Access Control Server (ACS).
- Cisco TrustSec supports IPv4 addressing only.
- SXP cannot use the management (mgmt 0) interface.
- You cannot enable Cisco TrustSec on interfaces in half-duplex mode.
- Clearing policies does not take affect immediately; it requires a flap to occur. In addition, the way policies are cleared depends on whether the SGT is static or dynamic. For a static SGT, the SGT is reset to 0 after the flap occurs. For dynamic SGT, the SGT is downloaded again from the RADIUS server after the flap occurs.
- Cisco TrustSec supports management switch virtual interfaces (SVIs), not routed SVIs.
- The 802.1X feature must be enabled before you enable the Cisco TrustSec feature. However, none of the 802.1X interface level features are available. The 802.1X feature is only used for the device to authenticate with RADIUS.
- RBACL is only implemented on bridged Ethernet traffic and cannot be enabled on a routing VLAN or routing interface.
- The determination of whether a peer is trusted or not and its capability to propagate SGTs on egress are made at the physical interface level.
- Cisco TrustedSec interface configurations on port channel members must be exactly the same. If a port channel member is inconsistent with the other port channel members, it will be error disabled.
- In a vPC domain, use the configuration synchronization mode (config-sync) to create switch profiles to ensure that the Cisco TrustSec configuration is synchronized between peers. If you configure the same vPC differently on two peer switches, traffic is treated differently.
- The maximum number of RBACL TCAM entries is 128, with 4 entries used by default, and the remaining 124 entries user-configurable.
- Cisco TrustSec is not supported on Layer 3 interfaces or Virtual Routing and Forwarding (VRF) interfaces.

- The **cts-manual**, **cts trusted mode**, and **no-propagate sgt** configurations must be consistent among all FEX ports or vEthernet ports on the same fabric port. If these configurations are inconsistent, the interfaces are err-disabled.
- The **cts-manual**, **sgt value**, **cts trusted mode**, and **no-propagate sgt** configurations must be consistent among all port channel members on the same port channel. If these configurations are inconsistent, the interfaces are err-disabled.

Default Settings For Cisco TrustSec

This table lists the default settings for Cisco TrustSec parameters.

Table 11: Default Cisco TrustSec Parameters Settings

Parameters	Default
Cisco TrustSec	Disabled
SXP	Disabled
SXP default password	None
SXP reconcile period	120 seconds (2 minutes)
SXP retry period	60 seconds (1 minute)
RBACL logging	Disabled
RBACL statistics	Disabled

Configuring Cisco TrustSec

This section provides information about the configuration tasks for Cisco TrustSec.

Enabling the Cisco TrustSec Feature

You must enable both the 802.1X feature and the Cisco TrustSec feature on the Cisco NX-OS device before you can configure Cisco TrustSec. However, none of the 802.1X interface level features are available. The 802.1X feature is only used for the device to authenticate with RADIUS.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters configuration mode.

	Command or Action	Purpose
Step 2	feature dot1x Example: switch(config)# feature dot1x	Enables the 802.1X feature.
Step 3	feature cts Example: switch(config)# feature cts	Enables the Cisco TrustSec feature.
Step 4	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 5	(Optional) show cts Example: switch# show cts	Displays the Cisco TrustSec configuration.
Step 6	(Optional) show feature Example: switch# show feature	Displays the enabled status for features.
Step 7	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring Cisco TrustSec Device Credentials

You must configure unique Cisco TrustSec credentials on each Cisco TrustSec-enabled Cisco NX-OS device in your network. Cisco TrustSec uses the password in the credentials for device authentication.



Note You must also configure the Cisco TrustSec credentials for the Cisco NX-OS device on the Cisco Secure ACS (see the documentation at the following URL:

http://www.cisco.com/en/US/products/sw/secursw/ps5338/products_installation_and_configuration_guides_list.html).

Before you begin

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters configuration mode.
Step 2	cts device-id name password password Example: switch(config)# cts device-id MyDevice1 password Cisc0321	Configures a unique device ID and password. The <i>name</i> argument has a maximum length of 32 characters and is case sensitive.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) show cts Example: switch# show cts	Displays the Cisco TrustSec configuration.
Step 5	(Optional) show cts environment Example: switch# show cts environment	Displays the Cisco TrustSec environment data.
Step 6	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#), on page 106

Configuring AAA for Cisco TrustSec

You can use Cisco Secure ACS for Cisco TrustSec authentication. You must configure RADIUS server groups and specify the default AAA authentication and authorization methods on one of the Cisco TrustSec-enabled Cisco NX-OS devices in your network cloud.



Note Only the Cisco Secure ACS supports Cisco TrustSec.

Configuring AAA on the Cisco TrustSec Cisco NX-OS Devices

This section describes how to configure AAA on the Cisco NX-OS device in your Cisco TrustSec network cloud.

Before you begin

Obtain the IPv4 address or hostname for the Cisco ACS.

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	radius-server host { <i>ipv4-address</i> <i>ipv6-address</i> <i>hostname</i> } key [0 7] <i>key</i> pac Example: switch(config)# radius-server host 10.10.1.1 key L1a0K2s9 pac	Configures a RADIUS server host with a key and PAC. The <i>hostname</i> argument is The <i>key</i> argument is alphanumeric, case sensitive, and has a maximum length of 63 characters. The 0 option indicates that the key is in clear text. The 7 option indicates that the key is encrypted. The default is clear text.
Step 3	(Optional) show radius-server Example: switch# show radius-server	Displays the RADIUS server configuration.
Step 4	aaa group server radius <i>group-name</i> Example: switch(config)# aaa group server radius Rad1 switch(config-radius)#	Specifies the RADIUS server group and enters RADIUS server group configuration mode.
Step 5	server { <i>ipv4-address</i> <i>ipv6-address</i> <i>hostname</i> } Example: switch(config-radius)# server 10.10.1.1	Specifies the RADIUS server host address.
Step 6	use-vrf <i>vrf-name</i> Example: switch(config-radius)# use-vrf management	Specifies the management VRF for the AAA server group. Note If you use the management VRF, no further configuration is necessary for the devices in the network cloud. If you use a different VRF, you must configure the devices with that VRF.
Step 7	exit Example: switch(config-radius)# exit switch(config)#	Exits RADIUS server group configuration mode.

	Command or Action	Purpose
Step 8	aaa authentication cts default group <i>group-name</i> Example: <pre>switch(config)# aaa authentication cts default group Rad1</pre>	Specifies the RADIUS server groups to use for Cisco TrustSec authentication.
Step 9	aaa authorization cts default group <i>group-name</i> Example: <pre>switch(config)# aaa authentication cts default group Rad1</pre>	Specifies the RADIUS server groups to use for Cisco TrustSec authorization.
Step 10	exit Example: <pre>switch(config)# exit switch#</pre>	Exits configuration mode.
Step 11	(Optional) show radius-server groups <i>[group-name]</i> Example: <pre>switch# show radius-server group rad1</pre>	Displays the RADIUS server group configuration.
Step 12	(Optional) show aaa authentication Example: <pre>switch# show aaa authentication</pre>	Displays the AAA authentication configuration.
Step 13	(Optional) show aaa authorization Example: <pre>switch# show aaa authorization</pre>	Displays the AAA authorization configuration.
Step 14	(Optional) show cts pacs Example: <pre>switch# show cts pacs</pre>	Displays the Cisco TrustSec PAC information.
Step 15	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Configuring AAA on Cisco TrustSec Nonseed Cisco NX-OS Devices](#)

Configuring Cisco TrustSec Authentication in Manual Mode

You can manually configure Cisco TrustSec on an interface if your Cisco NX-OS device does not have access to a Cisco Secure ACS. You must manually configure the interfaces on both ends of the connection.



Caution For the Cisco TrustSec manual mode configuration to take effect, you must enable and disable the interface, which disrupts traffic on the interface.

Before you begin

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface { ethernet vethernet }slot/port Example: <pre>switch(config)# interface ethernet 2/2 switch(config-if)#</pre>	Specifies an interface and enters interface configuration mode.
Step 3	cts manual Example: <pre>switch(config-if)# cts manual switch(config-if-cts-manual)#</pre>	Enters Cisco TrustSec manual configuration mode. Note You cannot enable Cisco TrustSec on interfaces in half-duplex mode.
Step 4	(Optional) policy dynamic identity peer-name Example: <pre>switch(config-if-cts-manual)# policy dynamic identity MyDevice2</pre>	Configures dynamic authorization policy download. The <i>peer-name</i> argument is the Cisco TrustSec device ID for the peer device. The peer name is case sensitive. Note Ensure that you have configured the Cisco TrustSec credentials and AAA for Cisco TrustSec. Note The policy dynamic and policy static commands are mutually exclusive. Only one can be applied at a time. To change from one to the other, you must use the no form of the command to remove the configuration before configuring the other command.

	Command or Action	Purpose
Step 5	(Optional) policy static sgt tag [trusted] Example: <pre>switch(config-if-cts-manual)# policy static sgt 0x2</pre>	Configures a static authorization policy. The <i>tag</i> argument is in hexadecimal format and the range is from 0x2 to 0xffef. The trusted keyword indicates that traffic coming on the interface with this SGT should not have its tag overridden. Note The policy dynamic and policy static commands are mutually exclusive. Only one can be applied at a time. To change from one to the other, you must use the no form of the command to remove the configuration before configuring the other command.
Step 6	exit Example: <pre>switch(config-if-cts-manual)# exit switch(config-if)#</pre>	Exits Cisco TrustSec manual configuration mode.
Step 7	shutdown Example: <pre>switch(config-if)# shutdown</pre>	Disables the interface.
Step 8	no shutdown Example: <pre>switch(config-if)# no shutdown</pre>	Enables the interface and enables Cisco TrustSec authentication on the interface.
Step 9	exit Example: <pre>switch(config-if)# exit switch(config)#</pre>	Exits interface configuration mode.
Step 10	(Optional) show cts interface all Example: <pre>switch# show cts interface all</pre>	Displays the Cisco TrustSec configuration for the interfaces.
Step 11	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

Configuring Pause Frame Encryption or Decryption for Cisco TrustSec on Interfaces

Pause frames are MAC control frames used for Ethernet flow control. The ports on some line cards encrypt and decrypt pause frames while the ports on other line cards do not have this ability. This disparity causes interoperability issues and causes the ports to discard or ignore the pause frames.

You can configure if the pause frames are to be encrypted or clear on individual interfaces. You must configure the interfaces on both ends of the connection but can do so using either dot1x or manual mode. If two ports are connected to form a CTS link and one is clear pause capable and the other is secure (encryption/decryption) pause capable, the pause frames must be sent in the clear across the link in order for them to be correctly sent and received.



Note F1 Series modules, F2 Series modules, and the N7K-M132XP-12(L) module support only clear pause frames. All other M1 Series modules support both secure (encrypted and decrypted) and clear pause frames.



Note You cannot enable Cisco TrustSec on interfaces in half-duplex mode. Use the **show interface** command to determine if an interface is configured for half-duplex mode.



Caution For the pause frame encryption or decryption configuration to take effect, you must enable and disable the interface, which disrupts traffic on the interface.

Before you begin

Ensure that you enabled Cisco TrustSec.

Ensure that you have enabled flow control on the interface using the **flowcontrol {send | receive}** command.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet slot/port Example: switch(config)# interface ethernet 2/2 switch(config-if)#	Specifies an interface and enters interface configuration mode.
Step 3	cts dot1x or cts manual Example:	Enters Cisco TrustSec dot1x or manual configuration mode.

	Command or Action	Purpose
	<code>switch(config-if)# cts dot1x</code> <code>switch(config-if-cts-dot1x)#</code>	Note You cannot enable Cisco TrustSec on interfaces in half-duplex mode.
Step 4	[no] encrypt pause-frame Example: <code>switch(config-if-cts-dot1x)# no encrypt</code> <code>pause-frame</code>	Configures pause frame encryption or decryption for Cisco TrustSec on the interface. When no encrypt pause-frame is configured, the pause frames are sent in the clear. When encrypt pause-frame is configured, pause frames are sent encrypted over the CTS link.
Step 5	exit Example: <code>switch(config-if-cts-dot1x)# exit</code> <code>switch(config-if)#</code>	Exits Cisco TrustSec dot1x or manual configuration mode.
Step 6	shutdown Example: <code>switch(config-if)# shutdown</code>	Disables the interface.
Step 7	no shutdown Example: <code>switch(config-if)# no shutdown</code>	Enables the interface and enables pause frame encryption or decryption for Cisco TrustSec on the interface.
Step 8	exit Example: <code>switch(config-if)# exit</code> <code>switch(config)#</code>	Exits interface configuration mode.
Step 9	(Optional) show cts interface all Example: <code>switch# show cts interface all</code>	Displays the Cisco TrustSec configuration for the interfaces.
Step 10	(Optional) copy running-config startup-config Example: <code>switch# copy running-config</code> <code>startup-config</code>	Copies the running configuration to the startup configuration.

Configuring SGACL Policies

This section provides information about the configuration tasks for SGACL policies.

SGACL Policy Configuration Process

Follow these steps to configure Cisco TrustSec SGACL policies:

Procedure

- Step 1** For Layer 2 interfaces, enable SGACL policy enforcement for the VLANs with Cisco TrustSec-enabled interfaces.
- Step 2** If you are not using AAA on a Cisco Secure ACS to download the SGACL policy configuration, manually configure the SGACL mapping and policies.

Enabling SGACL Policy Enforcement on VLANs

If you use SGACLs, you must enable SGACL policy enforcement in the VLANs that have Cisco TrustSec-enabled Layer 2 interfaces.



Note This operation cannot be performed on FCoE VLANs.

Before you begin

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters configuration mode.
Step 2	vlan <i>vlan-id</i> Example: <pre>switch(config)# vlan 10 switch(config-vlan)#</pre>	Specifies a VLAN and enters VLAN configuration mode.
Step 3	cts role-based enforcement Example: <pre>switch(config-vlan)# cts role-based enforcement</pre>	Enables Cisco TrustSec SGACL policy enforcement on the VLAN.
Step 4	exit Example: <pre>switch(config-vlan)# exit switch(config)#</pre>	Saves the VLAN configuration and exits VLAN configuration mode.
Step 5	(Optional) show cts role-based enable Example: <pre>switch(config)# show cts role-based enable</pre>	Displays the Cisco TrustSec SGACL enforcement configuration.

	Command or Action	Purpose
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

Manually Configuring Cisco TrustSec SGTs

You can manually configure unique Cisco TrustSec security group tags (SGTs) for the packets originating from this device.

Before you begin

Ensure that you have enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters configuration mode.
Step 2	cts sgt tag Example: switch(config)# cts sgt 0x00a2	Configures the SGT for packets sent from the device. The <i>tag</i> argument is a hexadecimal value in the format 0xhhhh . The range is from 0x2 to 0xffef.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) show cts environment-data Example: switch# show cts environment-data	Displays the Cisco TrustSec environment data information.
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

Manually Configuring IPv4-Address-to-SGACL SGT Mapping for a VLAN

You can manually configure an IPv4 address to SGACL SGT mapping on a VLAN so that the policies for that SGT are downloaded from the ACS server, or if you are using SXP mode, the SGT mapping is relayed to the listener.

Before you begin

Ensure that you enabled Cisco TrustSec.

Ensure that you enabled SGACL policy enforcement on the VLAN.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters configuration mode.
Step 2	vlan <i>vlan-id</i> Example: switch(config)# vlan 10 switch(config-vlan)#	Specifies a VLAN and enters VLAN configuration mode.
Step 3	cts role-based sgt-map <i>ipv4-address tag</i> Example: switch(config-vlan)# cts role-based sgt-map 10.10.1.1 100	Configures SGT mapping for the SGACL policies for the VLAN.
Step 4	exit Example: switch(config-vlan)# exit switch(config)#	Saves the VLAN configuration and exits VLAN configuration mode.
Step 5	(Optional) show cts role-based sgt-map Example: switch(config)# show cts role-based sgt-map	Displays the Cisco TrustSec SGACL SGT mapping configuration.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Enabling SGACL Policy Enforcement on VLANs](#) , on page 115

Enabling SGACL Policy Enforcement on VRFs

Manually Configuring IPv4-Address-to-SGACL SGT Mapping for a VRF

You can manually configure IPv4 address to SGACL SGT mapping on either a VRF if a Cisco Secure ACS is not available to download the SGACL policy configuration. You can use this feature if you do not have Cisco Secure ACS available on your Cisco NX-OS device. The IPv4-SGT mapping for VRF is useful for the SXP speaker.



Note The `cts role based enforcement` command is not supported on VRF.

Before you begin

Ensure that you enabled Cisco TrustSec.

Ensure that the Layer-3 module is enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters configuration mode.
Step 2	vrf context <i>vrf-name</i> Example: switch(config)# vrf accounting switch(config-vrf)#	Specifies a VRF and enters VRF configuration mode.
Step 3	cts role-based sgt-map <i>ipv4-address tag</i> Example: switch(config-vrf)# cts role-based sgt-map 10.10.1.1 100	Configures SGT mapping for the SGACL policies for the VLAN.
Step 4	exit Example: switch(config-vrf)# exit switch(config)#	Exits VRF configuration mode.
Step 5	(Optional) show cts role-based sgt-map Example: switch(config)# show cts role-based sgt-map	Displays the Cisco TrustSec SGACL SGT mapping configuration.
Step 6	(Optional) copy running-config startup-config Example:	Copies the running configuration to the startup configuration.

	Command or Action	Purpose
	<code>switch(config)# copy running-config startup-config</code>	

Manually Configuring SGACL Policies

You can manually configure SGACL policies on your Cisco NX-OS device if a Cisco Secure ACS is not available to download the SGACL policy configuration. You can also enable role-based access control list (RBACL) logging, which allows users to monitor specific types of packets exiting the Cisco NX-OS device.

Before you begin

Ensure that you have enabled Cisco TrustSec.

For Cisco TrustSec logging to function, you must enable Cisco TrustSec counters or statistics.

Ensure that you have enabled SGACL policy enforcement on the VLAN.

If you plan to enable RBACL logging, ensure that you have enabled RBACL policy enforcement on the VLAN.

If you plan to enable RBACL logging, ensure that you have set the logging level of CTS manager syslogs to 6 or less.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	cts role-based access-list <i>list-name</i> Example: <code>switch(config)# cts role-based</code> <code>access-list MySGACL</code> <code>switch(config-rbacl)#</code>	Specifies an SGACL and enters role-based access list configuration mode. The <i>list-name</i> argument value is alphanumeric, case sensitive, and has a maximum length of 32 characters.
Step 3	(Optional) {deny permit} all [log] Example: <code>switch(config-rbacl)# deny all log</code>	Denies or permits all traffic. Optionally, you can use the log keyword to specify that packets matching this configuration be logged.
Step 4	(Optional) {deny permit} icmp [log] Example: <code>switch(config-rbacl)# permit icmp</code>	Denies or permits Internet Control Message Protocol (ICMP) traffic. Optionally, you can use the log keyword to specify that packets matching this configuration be logged.
Step 5	(Optional) {deny permit} igmp [log] Example: <code>switch(config-rbacl)# deny igmp</code>	Denies or permits Internet Group Management Protocol (IGMP) traffic. Optionally, you can use the log keyword to specify that packets matching this configuration be logged.

	Command or Action	Purpose
Step 6	(Optional) {deny permit} ip [log] Example: switch(config-rbacl)# permit ip	Denies or permits IP traffic. Optionally, you can use the log keyword to specify that packets matching this configuration be logged.
Step 7	(Optional) {deny permit} tcp [{dst src} {{eq gt lt neq} port-number range port-number1 port-number2}] [log] Example: switch(config-rbacl)# deny tcp dst eq 100	Denies or permits TCP traffic. The default permits all TCP traffic. The range for the <i>port-number</i> , <i>port-number1</i> , and <i>port-number2</i> arguments is from 0 to 65535. Optionally, you can use the log keyword to specify that packets matching this configuration be logged.
Step 8	{deny permit} udp [{dst src} {{eq gt lt neq} port-number range port-number1 port-number2}] [log] Example: switch(config-rbacl)# permit udp src eq 1312	Denies or permits UDP traffic. The default permits all UDP traffic. The range for the <i>port-number</i> , <i>port-number1</i> , and <i>port-number2</i> arguments is from 0 to 65535. Optionally, you can use the log keyword to specify that packets matching this configuration be logged.
Step 9	exit Example: switch(config-rbacl)# exit switch(config)#	Exits role-based access-list configuration mode.
Step 10	cts role-based sgt {sgt-value any unknown} dgt {dgt-value any unknown} access-list list-name Example: switch(config)# cts role-based sgt 3 dgt 10 access-list MySGACL	Maps the SGT values to the SGACL. The <i>sgt-value</i> and <i>dgt-value</i> argument values range from 0 to 65519. Note You must create the SGACL before you can map SGTs to it.
Step 11	(Optional) show cts role-based access-list Example: switch(config)# show cts role-based access-list	Displays the Cisco TrustSec SGACL configuration.
Step 12	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Enabling SGACL Policy Enforcement on VLANs](#) , on page 115

[Enabling SGACL Policy Enforcement on VRFs](#)

Displaying the Downloaded SGACL Policies

After you configure the Cisco TrustSec device credentials and AAA, you can verify the Cisco TrustSec SGACL policies downloaded from the Cisco Secure ACS. The Cisco NX-OS software download the SGACL policies when it learns of a new SGT through authentication and authorization on an interface or from manual IPv4 address to SGACL SGT mapping.

Before you begin

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	show cts role-based access-list Example: switch# show cts role-based access-list	Displays Cisco TrustSec SGACLs, both downloaded from the Cisco Secure ACS and manually configured on the Cisco NX-OS device.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

Refreshing the Downloaded SGACL Policies

You can refresh the SGACL policies downloaded to the Cisco NX-OS device by the Cisco Secure ACS.

Before you begin

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	cts refresh role-based policy Example: switch# cts refresh policy	Refreshes the Cisco TrustSec SGACL policies from the Cisco Secure ACS.
Step 2	(Optional) show cts role-based policy Example: switch# show cts role-based policy	Displays the Cisco TrustSec SGACL policies.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

Enabling Statistics for RBACL

You can request a count of the number of packets that match role-based access control list (RBACL) policies. These statistics are collected per ACE.



Note RBACL statistics are lost only when the Cisco NX-OS device reloads or you deliberately clear the statistics.

Before you begin

Ensure that you have enabled Cisco TrustSec.

If you plan to enable RBACL statistics, ensure that you have enabled RBACL policy enforcement on the VLAN.

When you enable RBACL statistics, each policy requires one entry in the hardware. If you do not have enough space remaining in the hardware, an error message appears, and you are unable to enable the statistics.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] cts role-based counters enable Example: switch(config)# cts role-based counters enable	Enables or disables RBACL statistics. The default is disabled.
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.
Step 4	exit Example: switch(config)# exit switch#	Exits global configuration mode.
Step 5	(Optional) show cts role-based counters Example: switch# show cts role-based counters	Displays the configuration status of RBACL statistics and lists statistics for all RBACL policies.
Step 6	(Optional) clear cts role-based counters Example: switch# clear cts role-based counters	Clears the RBACL statistics so that all counters are reset to 0.

Clearing Cisco TrustSec SGACL Policies

You can clear the Cisco TrustSec SGACL policies.



Note Clearing policies does not take affect immediately; it requires a flap to occur. In addition, the way policies are cleared depends on whether the SGT is static or dynamic. For a static SGT, the SGT is reset to 0 after the flap occurs. For dynamic SGT, the SGT is downloaded again from the RADIUS server after the flap occurs.

Before you begin

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	(Optional) show cts role-based policy Example: switch# clear cts policy all	Displays the Cisco TrustSec RBACL policy configuration.
Step 2	clear cts policy {all peer <i>device-name</i> sgt <i>sgt-value</i> } Example: switch# clear cts policy all	Clear the policies for Cisco TrustSec connection information.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

Manually Configuring SXP

You can use the SGT Exchange Protocol (SXP) to propagate the SGTs across network devices that do not have hardware support for Cisco TrustSec. This section describes how to configure Cisco TrustSec SXP on Cisco NX-OS devices in your network.

Cisco TrustSec SXP Configuration Process

Follow these steps to manually configure Cisco TrustSec SXP:

Procedure

- Step 1** Enable the Cisco TrustSec feature.
- Step 2** Enable Cisco TrustSec SXP.
- Step 3** Configure SXP peer connections.

Note You cannot use the management (mgmt 0) connection for SXP.

Related Topics

[Enabling SGACL Policy Enforcement on VLANs](#) , on page 115

[Enabling SGACL Policy Enforcement on VRFs](#)

[Manually Configuring IPv4-Address-to-SGACL SGT Mapping for a VLAN](#), on page 117

[Manually Configuring SGACL Policies](#), on page 119

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Enabling Cisco TrustSec SXP](#) , on page 124

[Configuring Cisco TrustSec SXP Peer Connections](#), on page 124

Enabling Cisco TrustSec SXP

You must enable Cisco TrustSec SXP before you can configure peer connections.

Before you begin

Ensure that you enabled Cisco TrustSec.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters configuration mode.
Step 2	cts sxp enable Example: switch(config)# cts sxp enable	Enables SXP for Cisco TrustSec.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) show cts sxp Example: switch# show cts sxp	Displays the SXP configuration.
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

Configuring Cisco TrustSec SXP Peer Connections

You must configure the SXP peer connection on both of the devices. One device is the speaker and the other is the listener. When using password protection, make sure to use the same password on both ends.



Note If the default SXP source IP address is not configured and you do not specify the SXP source address in the connection, the Cisco NX-OS software derives the SXP source IP address from existing local IP addresses. The SXP source address could be different for each TCP connection initiated from the Cisco NX-OS device.



Note This Cisco Nexus switch supports SXP speaker mode only. Therefore, any SXP peer must be configured as a listener.

Before you begin

- Ensure that you enabled Cisco TrustSec.
- Ensure that you enabled SXP.
- Ensure that you enabled RBACL policy enforcement in the VRF.

Procedure

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>switch# configure terminal switch(config)#</pre>	Enters configuration mode.
Step 2	<p>cts sxp connection peer <i>peer-ipv4-addr</i> [source <i>src-ipv4-addr</i>] password {default none required <i>password</i>} mode listener [vrf <i>vrf-name</i>]</p> <p>Example:</p> <pre>switch(config)# cts sxp connection peer 10.10.1.1 source 20.20.1.1 password default mode listener</pre>	<p>Configures the SXP address connection.</p> <p>The source keyword specifies the IPv4 address of the source device. The default source is IPv4 address you configured using the cts sxp default source-ip command.</p> <p>The password keyword specifies the password that SXP should use for the connection using the following options:</p> <p>default Uses the default SXP password you configured using the cts sxp default password command.</p> <p>none Does not use a password.</p> <p>required Uses the password specified in the command.</p> <p>The vrf keyword specifies the VRF to the peer. The default is the default VRF.</p> <p>The mode listener keyword specifies the role of the remote peer device. Because this Cisco</p>

	Command or Action	Purpose
		Nexus Series switch can only act as the speaker in the connection, the peer must be configured as the listener. Note You cannot use the management (mgmt 0) interface for SXP.
Step 3	exit Example: switch(config)# exit switch#	Exits configuration mode.
Step 4	(Optional) show cts sxp connections Example: switch# show cts sxp connections	Displays the SXP connections and their status.
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Enabling Cisco TrustSec SXP](#) , on page 124

[Enabling SGACL Policy Enforcement on VRFs](#)

Configuring the Default SXP Password

By default, SXP uses no password when setting up connections. You can configure a default SXP password for the Cisco NX-OS device.

Before you begin

Ensure that you enabled Cisco TrustSec.

Ensure that you enabled SXP.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters configuration mode.
Step 2	cts sxp default password <i>password</i> Example:	Configures the SXP default password.

	Command or Action	Purpose
	<code>switch(config)# cts sxp default password A2Q3d4F5</code>	
Step 3	exit Example: <code>switch(config)# exit</code> <code>switch#</code>	Exits configuration mode.
Step 4	(Optional) show cts sxp Example: <code>switch# show cts sxp</code>	Displays the SXP configuration.
Step 5	(Optional) show running-config cts Example: <code>switch# show running-config cts</code>	Displays the SXP configuration in the running configuration.
Step 6	(Optional) copy running-config startup-config Example: <code>switch# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Enabling Cisco TrustSec SXP](#) , on page 124

Configuring the Default SXP Source IPv4 Address

The Cisco NX-OS software uses the default source IPv4 address in all new TCP connections where a source IPv4 address is not specified. There is no effect on existing TCP connections when you configure the default SXP source IPv4 address.

Before you begin

Ensure that you enabled Cisco TrustSec.

Ensure that you enabled SXP.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters configuration mode.
Step 2	cts sxp default source-ip <i>src-ip-addr</i> Example:	Configures the SXP default source IPv4 address.

	Command or Action	Purpose
	<code>switch(config)# cts sxp default source-ip 10.10.3.3</code>	
Step 3	exit Example: <code>switch(config)# exit</code> <code>switch#</code>	Exits configuration mode.
Step 4	(Optional) show cts sxp Example: <code>switch# show cts sxp</code>	Displays the SXP configuration.
Step 5	(Optional) copy running-config startup-config Example: <code>switch# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Enabling Cisco TrustSec SXP](#) , on page 124

Changing the SXP Retry Period

The SXP retry period determines how often the Cisco NX-OS software retries an SXP connection. When an SXP connection is not successfully set up, the Cisco NX-OS software makes a new attempt to set up the connection after the SXP retry period timer expires. The default value is 60 seconds (1 minute). Setting the SXP retry period to 0 seconds disables the timer and retries are not attempted.

Before you begin

Ensure that you enabled Cisco TrustSec.

Ensure that you enabled SXP.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters configuration mode.
Step 2	cts sxp retry-period <i>seconds</i> Example: <code>switch(config)# cts sxp retry-period 120</code>	Changes the SXP retry timer period. The default value is 60 seconds (1 minute). The range is from 0 to 64000.
Step 3	exit Example:	Exits configuration mode.

	Command or Action	Purpose
	switch(config)# exit switch#	
Step 4	(Optional) show cts sxp Example: switch# show cts sxp	Displays the SXP configuration.
Step 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Enabling the Cisco TrustSec Feature](#) , on page 106

[Enabling Cisco TrustSec SXP](#) , on page 124

Verifying the Cisco TrustSec Configuration

To display Cisco TrustSec configuration information, perform one of the following tasks:

Command	Purpose
show cts	Displays Cisco TrustSec information.
show cts credentials	Displays Cisco TrustSec credentials for EAP-FAST.
show cts environment-data	Displays Cisco TrustSec environmental data.
show cts interface	Displays the Cisco TrustSec configuration for the interfaces.
show cts role-based access-list	Displays Cisco TrustSec SGACL information.
show cts role-based counters	Displays the configuration status of RBACL statistics and lists statistics for all RBACL policies.
show cts role-based enable	Displays Cisco TrustSec SGACL enforcement status.
show cts role-based policy	Displays Cisco TrustSec SGACL policy information.
show cts role-based sgt-map	Displays Cisco TrustSec SGACL SGT map configuration.
show cts sxp	Displays Cisco TrustSec SXP information.
show running-config cts	Displays the Cisco TrustSec information in the running configuration.

Configuration Examples for Cisco TrustSec

This section provides configuration examples for Cisco TrustSec.

Enabling Cisco TrustSec

The following example shows how to enable Cisco TrustSec:

```
feature cts
cts device-id device1 password Cisco321
```

Configuring AAA for Cisco TrustSec on a Cisco NX-OS Device

The following example shows how to configure AAA for Cisco TrustSec on the Cisco NX-OS device:

```
radius-server host 10.10.1.1 key Cisco123 pac
aaa group server radius Rad1
  server 10.10.1.1
  use-vrf management
aaa authentication cts default group Rad1
aaa authorization cts default group Rad1
```

Configuring Cisco TrustSec Authentication in Manual Mode

The following example shows how to configure Cisco TrustSec authentication in manual mode static policy on an interface:

```
interface ethernet 2/1
  cts manual
  policy static sgt 0x20
  no propagate-sgt
```

The following example shows how to configure Cisco TrustSec authentication in manual mode dynamic policy on an interface:

```
interface ethernet 2/2
  cts manual
  policy dynamic identity device2
```

Configuring Cisco TrustSec Role-Based Policy Enforcement for a VLAN

The following example shows how to enable Cisco TrustSec role-based policy enforcement for a VLAN:

```
vlan 10
  cts role-based enforcement
```

Configuring IPv4 Address to SGACL SGT Mapping for the Default VRF

The following example shows how to manually configure IPv4 address to SGACL SGT mapping for Cisco TrustSec role-based policies for the default VRF:

```
cts role-based sgt-map 10.1.1.1 20
```

Configuring IPv4 Address to SGACL SGT Mapping for a VLAN

The following example shows how to manually configure IPv4 address to SGACL SGT mapping for Cisco TrustSec role-based policies for a VLAN:

```
vlan 10
  cts role-based sgt-map 20.1.1.1 20
```

Manually Configuring Cisco TrustSec SGACLs

The following example shows how to manually configure Cisco TrustSec SGACLs:

```
cts role-based access-list abcd
  permit icmp
cts role-based sgt 10 dgt 20 access-list abcd
```

The following example shows how to enable RBACL logging:

```
cts role-based access-list RBACL1
  deny tcp src eq 1111 dest eq 2222 log
cts role-based sgt 10 dgt 20 access-list RBACL1
```

The above configuration generates the following ACLLOG syslog:

```
%% VDC-1 %% %CTS-6-CTS_RBACL_STAT_LOG: CTS ACE permit all log, Threshold exceeded: Hit count
in 10s period = 4
```



Note The ACLLOG syslog does not contain the destination group tag (DGT) information of the matched RBACL policy.

The following example shows how to enable and display RBACL statistics:

```
cts role-based counters enable
show cts role-based counters

RBACL policy counters enabled
Counters last cleared: 06/08/2009 at 01:32:59 PM
rbacl:abc
  deny tcp dest neq 80 [0]
  deny tcp dest range 78 79 [0]
```

```

rbacl: def
    deny udp [0]
    deny ip [0]
    deny igmp [0]

```

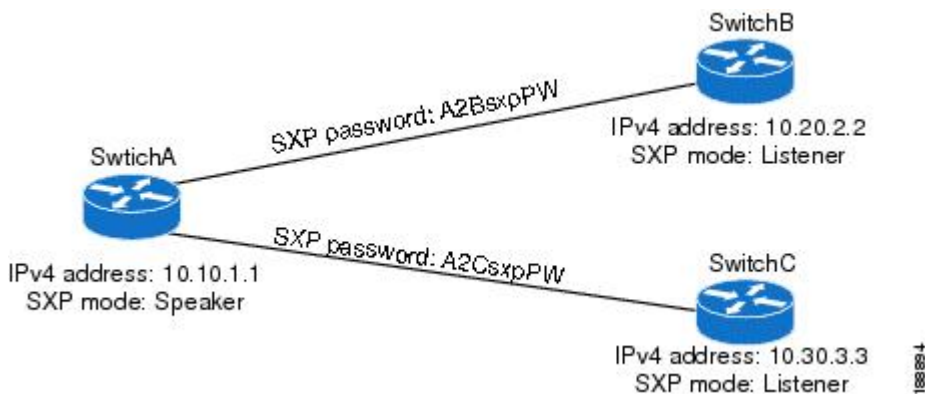
Manually Configuring SXP Peer Connections

This figure shows an example of SXP peer connections over the default VRF.

Figure 11: Example SXP Peer Connections



Note Because this Cisco Nexus switch supports only SXP speaker mode, it can only be configured as SwitchA in this example.



The following example shows how to configure the SXP peer connections on SwitchA:

```

feature cts
cts sxp enable
cts sxp connection peer 10.20.2.2 password required A2BsxpPW mode listener
cts sxp connection peer 10.30.3.3 password required A2CsxpPW mode listener

```

The following example shows how to configure the SXP peer connection on SwitchB:

```

feature cts
cts sxp enable
cts sxp connection peer 10.10.1.1 password required A2BsxpPW mode speaker

```

The following example shows how to configure the SXP peer connection on SwitchC:

```

feature cts
cts sxp enable
cts sxp connection peer 10.10.1.1 password required A2CsxpPW mode speaker

```

Additional References for Cisco TrustSec

This sections provides additional information related to implementing Cisco TrustSec.

Related Documentation

Related Topic	Document Title
Cisco NX-OS licensing	<i>Cisco NX-OS Licensing Guide</i>
Command Reference	<i>Cisco Nexus 5500 Series NX-OS TrustSec Command Reference</i>

Feature History for Cisco TrustSec

This table lists the release history for this feature.

Table 12: Feature History for Cisco TrustSec

Feature Name	Releases	Feature Information
Cisco TrustSec	5.1(3)N1(1)	This feature was introduced.



CHAPTER 9

Configuring Access Control Lists

This chapter contains the following sections:

- [Information About ACLs, on page 135](#)
- [Configuring IP ACLs, on page 142](#)
- [Configuring MAC ACLs, on page 149](#)
- [Example Configuration for MAC ACLs, on page 153](#)
- [Information About VLAN ACLs, on page 154](#)
- [Configuring VACLs, on page 154](#)
- [Configuration Examples for VACL, on page 157](#)
- [Configuring ACLs on Virtual Terminal Lines, on page 157](#)

Information About ACLs

An access control list (ACL) is an ordered set of rules that you can use to filter traffic. Each rule specifies a set of conditions that a packet must satisfy to match the rule. When the switch determines that an ACL applies to a packet, it tests the packet against the conditions of all rules. The first match determines whether the packet is permitted or denied. If there is no match, the switch applies the applicable default rule. The switch continues processing packets that are permitted and drops packets that are denied.

You can use ACLs to protect networks and specific hosts from unnecessary or unwanted traffic. For example, you could use ACLs to disallow HTTP traffic from a high-security network to the Internet. You could also use ACLs to allow HTTP traffic but only to specific sites, using the IP address of the site to identify it in an IP ACL.

IP ACL Types and Applications

The Cisco Nexus device supports IPv4, IPv6, and MAC ACLs for security traffic filtering. The switch allows you to use IP access control lists (ACLs) as port ACLs, VLAN ACLs, and Router ACLs as shown in the following table.

Table 13: Security ACL Applications

Application	Supported Interfaces	Types of ACLs Supported
Port ACL	<p>An ACL is considered a port ACL when you apply it to one of the following:</p> <ul style="list-style-type: none"> • Ethernet interface • Ethernet port-channel interface <p>When a port ACL is applied to a trunk port, the ACL filters traffic on all VLANs on the trunk port.</p>	IPv4 ACLs IPv6 ACLs MAC ACLs
Router ACL	<ul style="list-style-type: none"> • VLAN interfaces <p>Note You must enable VLAN interfaces globally before you can configure a VLAN interface.</p> <ul style="list-style-type: none"> • Physical Layer 3 interfaces • Layer 3 Ethernet subinterfaces • Layer 3 Ethernet port-channel interfaces • Layer 3 Ethernet port-channel subinterfaces • Tunnels • Management interfaces 	IPv4 ACLs IPv6 ACLs
VLAN ACL (VACL)	<p>An ACL is a VACL when you use an access map to associate the ACL with an action and then apply the map to a VLAN.</p>	IPv4 ACLs MAC ACLs
VTY ACL	VTYs	IPv4 ACLs IPv6 ACLs

Application Order

When the device processes a packet, it determines the forwarding path of the packet. The path determines which ACLs that the device applies to the traffic. The device applies the ACLs in the following order:

1. Port ACL
2. Ingress VACL
3. Ingress Router ACL
4. Egress Router ACL
5. Egress VACL

Rules

You can create rules in access-list configuration mode by using the **permit** or **deny** command. The switch allows traffic that matches the criteria in a permit rule and blocks traffic that matches the criteria in a deny rule. You have many options for configuring the criteria that traffic must meet in order to match the rule.

Source and Destination

In each rule, you specify the source and the destination of the traffic that matches the rule. You can specify both the source and destination as a specific host, a network or group of hosts, or any host.

Protocols

IPv4, IPv6, and MAC ACLs allow you to identify traffic by protocol. For your convenience, you can specify some protocols by name. For example, in an IPv4 ACL, you can specify ICMP by name.

You can specify any protocol by the integer that represents the Internet protocol number. For example, you can use 115 to specify Layer 2 Tunneling Protocol (L2TP) traffic.

Implicit Rules

IP and MAC ACLs have implicit rules, which means that although these rules do not appear in the running configuration, the switch applies them to traffic when no other rules in an ACL match.

All IPv4 ACLs include the following implicit rule:

```
deny ip any any
```

This implicit rule ensures that the switch denies unmatched IP traffic.

All IPv6 ACLs include the following implicit rule:

```
deny ipv6 any any
```

Additional Filtering Options

You can identify traffic by using additional options. IPv4 ACLs support the following additional filtering options:

- Layer 4 protocol
- TCP and UDP ports
- ICMP types and codes
- IGMP types
- Precedence level
- Differentiated Services Code Point (DSCP) value
- TCP packets with the ACK, FIN, PSH, RST, SYN, or URG bit set
- Established TCP connections

IPv6 ACLs support the following additional filtering options:

- Layer 4 protocol

- Authentication Header Protocol
- Encapsulating Security Payload
- Payload Compression Protocol
- Stream Control Transmission Protocol (SCTP)
- SCTP, TCP, and UDP ports
- ICMP types and codes
- IGMP types
- Flow label
- DSCP value
- TCP packets with the ACK, FIN, PSH, RST, SYN, or URG bit set
- Established TCP connections
- Packet length

MAC ACLs support the following additional filtering options:

- Layer 3 protocol
- VLAN ID
- Class of Service (CoS)

Sequence Numbers

The Cisco Nexus device supports sequence numbers for rules. Every rule that you enter receives a sequence number, either assigned by you or assigned automatically by the device. Sequence numbers simplify the following ACL tasks:

- Adding new rules between existing rules—By specifying the sequence number, you specify where in the ACL a new rule should be positioned. For example, if you need to insert a rule between rules numbered 100 and 110, you could assign a sequence number of 105 to the new rule.
- Removing a rule—Without using a sequence number, removing a rule requires that you enter the whole rule, as follows:

```
switch(config-acl)# no permit tcp 10.0.0.0/8 any
```

However, if the same rule had a sequence number of 101, removing the rule requires only the following command:

```
switch(config-acl)# no 101
```

- Moving a rule—With sequence numbers, if you need to move a rule to a different position within an ACL, you can add a second instance of the rule using the sequence number that positions it correctly, and then you can remove the original instance of the rule. This action allows you to move the rule without disrupting traffic.

If you enter a rule without a sequence number, the device adds the rule to the end of the ACL and assigns a sequence number that is 10 greater than the sequence number of the preceding rule to the rule. For example,

if the last rule in an ACL has a sequence number of 225 and you add a rule without a sequence number, the device assigns the sequence number 235 to the new rule.

In addition, the device allows you to reassign sequence numbers to rules in an ACL. Resequencing is useful when an ACL has rules numbered contiguously, such as 100 and 101, and you need to insert one or more rules between those rules.

Logical Operators and Logical Operation Units

IP ACL rules for TCP and UDP traffic can use logical operators to filter traffic based on port numbers.

The Cisco Nexus device stores operator-operand couples in registers called logical operation units (LOUs) to perform operations (greater than, less than, not equal to, and range) on the TCP and UDP ports specified in an IP ACL.



Note The range operator is inclusive of boundary values.

These LOUs minimize the number of ternary content addressable memory (TCAM) entries needed to perform these operations. A maximum of two LOUs are allowed for each feature on an interface. For example an ingress RACL can use two LOUs, and a QoS feature can use two LOUs. If an ACL feature requires more than two arithmetic operations, the first two operations use LOUs, and the remaining access control entries get expanded.

The following guidelines determine when the device stores operator-operand couples in LOUs:

- If the operator or operand differs from other operator-operand couples that are used in other rules, the couple is stored in an LOU.

For example, the operator-operand couples "gt 10" and "gt 11" would be stored separately in half an LOU each. The couples "gt 10" and "lt 10" would also be stored separately.

- Whether the operator-operand couple is applied to a source port or a destination port in the rule affects LOU usage. Identical couples are stored separately when one of the identical couples is applied to a source port and the other couple is applied to a destination port.

For example, if a rule applies the operator-operand couple "gt 10" to a source port and another rule applies a "gt 10" couple to a destination port, both couples would also be stored in half an LOU, resulting in the use of one whole LOU. Any additional rules using a "gt 10" couple would not result in further LOU usage.

Statistics and ACLs

The device can maintain global statistics for each rule that you configure in IPv4, IPv6, and MAC ACLs. If an ACL is applied to multiple interfaces, the maintained rule statistics are the sum of packet matches (hits) on all the interfaces on which that ACL is applied.



Note The device does not support interface-level ACL statistics.

For each ACL that you configure, you can specify whether the device maintains statistics for that ACL, which allows you to turn ACL statistics on or off as needed to monitor traffic filtered by an ACL or to help troubleshoot the configuration of an ACL.

The device does not maintain statistics for implicit rules in an ACL. For example, the device does not maintain a count of packets that match the implicit **deny ip any any** rule at the end of all IPv4 ACLs. If you want to maintain statistics for implicit rules, you must explicitly configure the ACL with rules that are identical to the implicit rules.

Licensing Requirements for ACLs

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	No license is required to use ACLs.

Prerequisites for ACLs

IP ACLs have the following prerequisites:

- You must be familiar with IP addressing and protocols to configure IP ACLs.
- You must be familiar with the interface types that you want to configure with ACLs.

VACLs have the following prerequisite:

- Ensure that the IP ACL or MAC ACL that you want to use in the VACL exists and is configured to filter traffic in the manner that you need for this application.

Guidelines and Limitations for ACLs

IP ACLs have the following configuration guidelines and limitations:

- We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. This is especially useful for ACLs that include more than about 1000 rules.
- When you apply an ACL that uses time ranges, the device updates the ACL entries whenever a time range referenced in an ACL entry starts or ends. Updates that are initiated by time ranges occur on a best-effort priority. If the device is especially busy when a time range causes an update, the device may delay the update by up to a few seconds.
- To apply an IP ACL to a VLAN interface, you must have enabled VLAN interfaces globally.

MAC ACLs have the following configuration guidelines and limitations:

- MAC ACLs apply to ingress traffic only.
- ACL statistics are not supported if the DHCP snooping feature is enabled.

- For M1 Series modules, the **mac packet-classify** command enables a MAC ACL for port and VLAN policies.

VACLs have the following configuration guidelines:

- We recommend that you perform ACL configurations using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration.
- ACL statistics are not supported if the DHCP snooping feature is enabled.

Default ACL Settings

The following table lists the default settings for IP ACLs parameters.

Table 14: Default IP ACLs Parameters

Parameters	Default
IP ACLs	No IP ACLs exist by default.
ACL rules	Implicit rules apply to all ACLs

The following table lists the default settings for MAC ACLs parameters.

Table 15: Default MAC ACLs Parameters

Parameters	Default
MAC ACLs	No MAC ACLs exist by default.
ACL rules	Implicit rules apply to all ACLs

The following table lists the default settings for VACL parameters.

Table 16: Default VACL Parameters

Parameters	Default
VACLs	No IP ACLs exist by default.
ACL rules	Implicit rules apply to all ACLs.

Configuring IP ACLs

Creating an IP ACL

You can create an IPv4 or IPv6 ACL on the switch and add rules to it.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# {ip ipv6} access-list name	Creates the IP ACL and enters IP ACL configuration mode. The <i>name</i> argument can be up to 64 characters.
Step 3	switch(config-acl)# [<i>sequence-number</i>] {permit deny} protocol source destination	Creates a rule in the IP ACL. You can create many rules. The <i>sequence-number</i> argument can be a whole number between 1 and 4294967295. The permit and deny commands support many ways of identifying traffic. For more information, see the <i>Command Reference</i> for the specific Cisco Nexus device.
Step 4	(Optional) switch(config-acl)# statistics	Specifies that the switch maintains global statistics for packets that matches the rules in the ACL.
Step 5	(Optional) switch# show {ip ipv6} access-lists name	Displays the IP ACL configuration.
Step 6	(Optional) switch# show ip access-lists name	Displays the IP ACL configuration.
Step 7	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to create an IPv4 ACL:

```
switch# configure terminal
switch(config)# ip access-list acl-01
switch(config-acl)# permit ip 192.168.2.0/24 any
switch(config-acl)# statistics
```

The following example shows how to create an IPv6 ACL:

```
switch# configure terminal
switch(config)# ipv6 access-list acl-01-ipv6
switch(config-ipv6-acl)# permit tcp 2001:0db8:85a3::/48 2001:0db8:be03:2112::/64
```


Changing an IP ACL

You can add and remove rules in an existing IPv4 or IPv6 ACL. You cannot change existing rules. Instead, to change a rule, you can remove it and recreate it with the desired changes.

If you need to add more rules between existing rules than the current sequence numbering allows, you can use the **resequence** command to reassign sequence numbers.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# {ip ipv6} access-list name	Enters IP ACL configuration mode for the ACL that you specify by name.
Step 3	switch(config)# ip access-list name	Enters IP ACL configuration mode for the ACL that you specify by name.
Step 4	switch(config-acl)# [<i>sequence-number</i>] {permit deny} protocol source destination	Creates a rule in the IP ACL. Using a sequence number allows you to specify a position for the rule in the ACL. Without a sequence number, the rule is added to the end of the rules. The <i>sequence-number</i> argument can be a whole number between 1 and 4294967295. The permit and deny commands support many ways of identifying traffic. For more information, see the <i>Command Reference</i> for your Cisco Nexus device.
Step 5	(Optional) switch(config-acl)# no {sequence-number {permit deny} protocol source destination}	Removes the rule that you specified from the IP ACL. The permit and deny commands support many ways of identifying traffic. For more information, see the <i>Command Reference</i> for your Cisco Nexus device.
Step 6	(Optional) switch(config-acl)# [no] statistics	Specifies that the switch maintains global statistics for packets matching the rules in the ACL. The no option stops the switch from maintaining global statistics for the ACL.
Step 7	(Optional) switch# show ip access-lists name	Displays the IP ACL configuration.
Step 8	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Changing Sequence Numbers in an IP ACL](#), on page 144

Removing an IP ACL

You can remove an IP ACL from the switch.

Before you remove an IP ACL from the switch, be sure that you know whether the ACL is applied to an interface. The switch allows you to remove ACLs that are currently applied. Removing an ACL does not affect the configuration of interfaces where you have applied the ACL. Instead, the switch considers the removed ACL to be empty.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# no {ip ipv6} access-list name	Removes the IP ACL that you specified by name from the running configuration.
Step 3	switch(config)# no ip access-list name	Removes the IP ACL that you specified by name from the running configuration.
Step 4	(Optional) switch# show running-config	Displays the ACL configuration. The removed IP ACL should not appear.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Changing Sequence Numbers in an IP ACL

You can change all the sequence numbers assigned to the rules in an IP ACL.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# resequence {ip ipv6} access-list name starting-sequence-number increment	Assigns sequence numbers to the rules contained in the ACL, where the first rule receives the starting sequence number that you specify. Each subsequent rule receives a number larger than the preceding rule. The difference in numbers is determined by the increment that you specify. The <i>starting-sequence-number</i> argument and the <i>increment</i> argument can be a whole number between 1 and 4294967295.
Step 3	(Optional) switch# show {ip ipv6} access-lists name	Displays the IP ACL configuration.
Step 4	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring ACLs with Logging

You can create an access-control list for logging traffic of a specified protocol and address.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# {ip ipv6} access-list name	Creates the IP ACL and enters IP ACL configuration mode. The <i>name</i> argument can be up to 64 characters.
Step 3	switch(config-acl)# permit protocol source destination log	<p>Creates a rule to log traffic of the specified protocol in the syslog file. in the IP ACL. Valid values for the <i>protocol</i> argument are:</p> <ul style="list-style-type: none"> • icmp—ICMP • igmp—IGMP • ip—IPv4 • ipv6—IPv6 • tcp—TCP • udp—UDP • sctp—SCTP (IPv6 only) <p>The source and destination arguments can be the IP address with a network wildcard (IPv4 only), IP address and variable-length subnet mask, host address, or any to designate any address. For more information, see the System Management configuration guide and the Security command reference for your platform.</p>
Step 4	switch(config-acl)# exit	Exits the current configuration mode.
Step 5	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to create an ACL for logging entries that match IPv4 TCP traffic from any source and any destination:

```
switch# configuration terminal
switch(config)# ip access-list tcp_log
switch(config-acl)# permit tcp any any log
```

```
switch(config-acl)# exit
switch(config)# copy running-config startup-config
```

Applying an IP ACL to mgmt0

You can apply an IPv4 or IPv6 ACL to the management interface (mgmt0).

Before you begin

Ensure that the ACL that you want to apply exists and that it is configured to filter traffic in the manner that you need for this application.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface mgmt port Example: switch(config)# interface mgmt0 switch(config-if)#	Enters configuration mode for the management interface.
Step 3	ip access-group access-list {in out} Example: switch(config-if)#ip access-group acl-120 out	Applies an IPv4 or IPv6 ACL to the Layer 3 interface for traffic flowing in the direction specified. You can apply one router ACL per direction.
Step 4	(Optional) show running-config aclmgr Example: switch(config-if)# show running-config aclmgr	Displays the ACL configuration.
Step 5	(Optional) copy running-config startup-config Example: switch(config-if)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

- Creating an IP ACL

Applying an IP ACL as a Router ACL

You can apply an IPv4 or IPv6 ACL to any of the following types of interfaces:

- Physical Layer 3 interfaces and subinterfaces
- Layer 3 Ethernet port-channel interfaces and subinterfaces
- VLAN interfaces
- Tunnels
- Management interfaces

ACLs applied to these interface types are considered router ACLs.

Before you begin

Ensure that the ACL you want to apply exists and that it is configured to filter traffic in the manner that you need for this application.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • switch(config)# interface ethernet <i>slot/port</i> [. <i>number</i>] • switch(config)# interface port-channel <i>channel-number</i> [. <i>number</i>] • switch(config)# interface tunnel <i>tunnel-number</i> • switch(config)# interface vlan <i>vlan-ID</i> • switch(config)# interface mgmt <i>port</i> 	Enters configuration mode for the interface type that you specified.
Step 3	Enter one of the following commands: <ul style="list-style-type: none"> • switch(config-if)# ip access-group <i>access-list</i> {in out} • switch(config-if)# ipv6 traffic-filter <i>access-list</i> {in out} 	Applies an IPv4 or IPv6 ACL to the Layer 3 interface for traffic flowing in the direction specified. You can apply one router ACL per direction.
Step 4	(Optional) switch(config-if)# show running-config aclmgr	Displays the ACL configuration.
Step 5	(Optional) switch(config-if)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Applying an IP ACL as a Port ACL

You can apply an IPv4 or IPv6 ACL to a physical Ethernet interface or a PortChannel. ACLs applied to these interface types are considered port ACLs.



Note Some configuration parameters when applied to an PortChannel are not reflected on the configuration of the member ports.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# interface { ethernet [chassis/]slot/port port-channel channel-number}	Enters interface configuration mode for the specified interface.
Step 3	switch(config-if)# { ip port access-group ipv6 port traffic-filter } <i>access-list in</i>	Applies an IPv4 or IPv6 ACL to the interface or PortChannel. Only inbound filtering is supported with port ACLs. You can apply one port ACL to an interface.
Step 4	(Optional) switch# show running-config	Displays the ACL configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying IP ACL Configurations

To display IP ACL configuration information, perform one of the following tasks:

Procedure

- switch# **show running-config**
Displays ACL configuration, including IP ACL configuration and interfaces that IP ACLs are applied to.
- switch# **show running-config interface**
Displays the configuration of an interface to which you have applied an ACL.

Example

For detailed information about the fields in the output from these commands, refer to the *Command Reference* for your Cisco Nexus device.

Monitoring and Clearing IP ACL Statistics

Use the **show ip access-lists** or **show ipv6 access-list** command to display statistics about an IP ACL, including the number of packets that have matched each rule. For detailed information about the fields in the output from this command, see the *Command Reference* for your Cisco Nexus device.



Note The mac access-list is applicable to non-IPv4 and non-IPv6 traffic only.

Procedure

- switch# **show {ip | ipv6} access-lists name**

Displays IP ACL configuration. If the IP ACL includes the **statistics** command, then the **show ip access-lists** and **show ipv6 access-list** command output includes the number of packets that have matched each rule.

- switch# **show ip access-lists name**

Displays IP ACL configuration. If the IP ACL includes the **statistics** command, then the **show ip access-lists** command output includes the number of packets that have matched each rule.

- switch# **clear {ip | ipv6} access-list counters [access-list-name]**

Clears statistics for all IP ACLs or for a specific IP ACL.

- switch# **clear ip access-list counters [access-list-name]**

Clears statistics for all IP ACLs or for a specific IP ACL.

Configuring MAC ACLs

Creating a MAC ACL

To create a MAC ACL and add rules to it, perform this task:

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch# mac access-list name	Creates the MAC ACL and enters ACL configuration mode.
Step 3	switch(config-mac-acl)# <i>[sequence-number]</i> {permit deny} source destination protocol	Creates a rule in the MAC ACL. The permit and deny options support many ways of identifying traffic. For more information, see the Security command reference for your platform.
Step 4	(Optional) switch(config-mac-acl)# statistics	Specifies that the switch maintains global statistics for packets matching the rules in the ACL.
Step 5	(Optional) switch# show mac access-lists name	Displays the MAC ACL configuration.

	Command or Action	Purpose
Step 6	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to create a MAC ACL and add rules to it:

```
switch# configure terminal
switch(config)# mac access-list acl-mac-01
switch(config-mac-acl)# permit 00c0.4f00.0000 0000.00ff.ffff any
switch(config-mac-acl)# statistics
```

Changing a MAC ACL

In an existing MAC ACL, you can add and remove rules. You cannot change existing rules. Instead, to change a rule, you can remove it and recreate it with the desired changes.

If you need to add more rules between existing rules than the current sequence numbering allows, you can use the **resequence** command to reassign sequence numbers.

To change a MAC ACL, perform this task:

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# mac access-list name	Enters ACL configuration mode for the ACL that you specify by name.
Step 3	switch(config-mac-acl)# [<i>sequence-number</i>] {permit deny} source destination protocol	Creates a rule in the MAC ACL. Using a sequence number allows you to specify a position for the rule in the ACL. Without a sequence number, the rule is added to the end of the rules. The permit and deny commands support many ways of identifying traffic.
Step 4	(Optional) switch(config-mac-acl)# no {sequence-number {permit deny} source destination protocol}	Removes the rule that you specify from the MAC ACL. The permit and deny commands support many ways of identifying traffic.
Step 5	(Optional) switch(config-mac-acl)# [no] statistics	Specifies that the switch maintains global statistics for packets matching the rules in the ACL. The no option stops the switch from maintaining global statistics for the ACL.

	Command or Action	Purpose
Step 6	(Optional) switch# show mac access-lists name	Displays the MAC ACL configuration.
Step 7	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to change a MAC ACL:

```
switch# configure terminal
switch(config)# mac access-list acl-mac-01
switch(config-mac-acl)# 100 permit mac 00c0.4f00.00 0000.00ff.ffff any
switch(config-mac-acl)# statistics
```

Removing a MAC ACL

You can remove a MAC ACL from the switch.

Be sure that you know whether the ACL is applied to an interface. The switch allows you to remove ACLs that are current applied. Removing an ACL does not affect the configuration of interfaces where you have applied the ACL. Instead, the switch considers the removed ACL to be empty.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# no mac access-list name	Removes the MAC ACL that you specify by name from the running configuration.
Step 3	(Optional) switch# show mac access-lists	Displays the MAC ACL configuration.
Step 4	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Changing Sequence Numbers in a MAC ACL

You can change all the sequence numbers assigned to rules in a MAC ACL. Resequencing is useful when you need to insert rules into an ACL and there are not enough available sequence numbers.

To change all the sequence numbers assigned to rules in a MAC ACL, perform this task:

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# resequence mac access-list <i>name starting-sequence-number increment</i>	Assigns sequence numbers to the rules contained in the ACL, where the first rule receives the number specified by the starting-sequence number that you specify. Each subsequent rule receives a number larger than the preceding rule. The difference in numbers is determined by the increment number that you specify.
Step 3	(Optional) switch# show mac access-lists <i>name</i>	Displays the MAC ACL configuration.
Step 4	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Rules](#), on page 137

Applying a MAC ACL as a Port ACL

You can apply a MAC ACL as a port ACL to any of the following interface types:

- Ethernet interfaces
- EtherChannel interfaces

Be sure that the ACL that you want to apply exists and is configured to filter traffic as necessary for this application.



Note Some configuration parameters when applied to an EtherChannel are not reflected on the configuration of the member ports.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# interface { ethernet <i>[chassis/]slot/port port-channel</i> <i>channel-number</i> }	Enters interface configuration mode for the Ethernet specified interface.
Step 3	switch(config-if)# mac port access-group <i>access-list</i>	Applies a MAC ACL to the interface.
Step 4	(Optional) switch# show running-config	Displays ACL configuration.
Step 5	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Related Topics

[Creating an IP ACL](#), on page 142

Verifying MAC ACL Configurations

To display MAC ACL configuration information, perform one of the following tasks:

Procedure

- switch# **show mac access-lists**
Displays the MAC ACL configuration
- switch# **show running-config**
Displays ACL configuration, including MAC ACLs and the interfaces that ACLs are applied to.
- switch# **show running-config interface**
Displays the configuration of the interface to which you applied the ACL.

Displaying and Clearing MAC ACL Statistics

Use the **show mac access-lists** command to display statistics about a MAC ACL, including the number of packets that have matched each rule.

Procedure

- switch# **show mac access-lists**
Displays MAC ACL configuration. If the MAC ACL includes the **statistics** command, the **show mac access-lists** command output includes the number of packets that have matched each rule.
- switch# **clear mac access-list counters**
Clears statistics for all MAC ACLs or for a specific MAC ACL.

Example Configuration for MAC ACLs

This example shows how to create a MAC ACL named `acl-mac-01` and apply it to Ethernet interface 1/1:

```
switch# configure terminal
switch(config)# mac access-list acl-mac-01
switch(config-mac-acl)# permit 00c0.4f00.0000 0000.00ff.ffff any
switch(config-mac-acl)# exit
switch(config)# interface ethernet 1/1
switch(config-if)# mac access-group acl-mac-01
```

Information About VLAN ACLs

A VLAN ACL (VACL) is one application of a MAC ACL or IP ACL. You can configure VACLs to apply to all packets that are bridged within a VLAN. VACLs are used strictly for security packet filtering. VACLs are not defined by direction (ingress or egress).

VACLs and Access Maps

VACLs use access maps to link an IP ACL or a MAC ACL to an action. The switch takes the configured action on packets permitted by the VACL.

VACLs and Actions

In access map configuration mode, you use the **action** command to specify one of the following actions:

- Forward—Sends the traffic to the destination determined by normal operation of the switch.
- Drop—Drops the traffic.

Statistics

The Cisco Nexus device can maintain global statistics for each rule in a VACL. If a VACL is applied to multiple VLANs, the maintained rule statistics are the sum of packet matches (hits) on all the interfaces on which that VACL is applied.



Note The Cisco Nexus device does not support interface-level VACL statistics.

For each VLAN access map that you configure, you can specify whether the switch maintains statistics for that VACL. This allows you to turn VACL statistics on or off as needed to monitor traffic filtered by a VACL or to help troubleshoot VLAN access-map configuration.

Configuring VACLs

Creating or Changing a VACL

You can create or change a VACL. Creating a VACL includes creating an access map that associates an IP ACL or MAC ACL with an action to be applied to the matching traffic.

To create or change a VACL, perform this task:

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# vlan access-map <i>map-name</i>	Enters access map configuration mode for the access map specified.
Step 3	switch(config-access-map)# match ip address <i>ip-access-list</i>	Specifies an IPv4 and IPv6 ACL for the map.
Step 4	switch(config-access-map)# match mac address <i>mac-access-list</i>	Specifies a MAC ACL for the map.
Step 5	switch(config-access-map)# action { drop forward }	Specifies the action that the switch applies to traffic that matches the ACL.
Step 6	(Optional) switch(config-access-map)# [no] statistics	Specifies that the switch maintains global statistics for packets matching the rules in the VACL. The no option stops the switch from maintaining global statistics for the VACL.
Step 7	(Optional) switch(config-access-map)# show running-config	Displays ACL configuration.
Step 8	(Optional) switch(config-access-map)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Removing a VACL

You can remove a VACL, which means that you will delete the VLAN access map.

Be sure that you know whether the VACL is applied to a VLAN. The switch allows you to remove VACLs that are current applied. Removing a VACL does not affect the configuration of VLANs where you have applied the VACL. Instead, the switch considers the removed VACL to be empty.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# no vlan access-map <i>map-name</i>	Removes the VLAN access map configuration for the specified access map.
Step 3	(Optional) switch(config)# show running-config	Displays ACL configuration.
Step 4	(Optional) switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Applying a VACL to a VLAN

You can apply a VACL to a VLAN.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# [no] vlan filter <i>map-name</i> vlan-list <i>list</i>	Applies the VACL to the VLANs by the list that you specified. The no option unapplies the VACL. The vlan-list command can specify a list of up to 32 VLANs, but multiple vlan-list commands can be configured to cover more than 32 VLANs.
Step 3	(Optional) switch(config)# show running-config	Displays ACL configuration.
Step 4	(Optional) switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying VACL Configuration

To display VACL configuration information, perform one of the following tasks:

Procedure

- switch# **show running-config aclmgr**
Displays ACL configuration, including VACL-related configuration.
- switch# **show vlan filter**
Displays information about VACLs that are applied to a VLAN.
- switch# **show vlan access-map**
Displays information about VLAN access maps.

Displaying and Clearing VACL Statistics

To display or clear VACL statistics, perform one of the following tasks:

Procedure

- switch# **show vlan access-list**
Displays VACL configuration. If the VLAN access-map includes the **statistics** command, then the **show vlan access-list** command output includes the number of packets that have matched each rule.
- switch# **clear vlan access-list counters**
Clears statistics for all VACLs or for a specific VACL.

Configuration Examples for VACL

The following example shows how to configure a VACL to forward traffic permitted by an IP ACL named `acl-ip-01` and how to apply the VACL to VLANs 50 through 82:

```
switch# configure terminal
switch(config)# vlan access-map acl-ip-map
switch(config-access-map)# match ip address acl-ip-01
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# vlan filter acl-ip-map vlan-list 50-82
```

Configuring ACLs on Virtual Terminal Lines

To restrict incoming and outgoing connections for IPv4 or IPv6 between a Virtual Terminal (VTY) line and the addresses in an access list, use the **access-class** command in line configuration mode. To remove access restrictions, use the **no** form of this command.

Follow these guidelines when configuring ACLs on VTY lines:

- Set identical restrictions on all VTY lines because a user can connect to any of them.
- Statistics per entry is not supported for ACLs on VTY lines.

Before you begin

Be sure that the ACL that you want to apply exists and is configured to filter traffic for this application.

Procedure

	Command or Action	Purpose
Step 1	<code>switch# configure terminal</code>	Enters global configuration mode.
Step 2	<code>switch(config)# line vty</code> Example: <code>switch(config)# line vty</code> <code>switch(config-line)#</code>	Enters line configuration mode.
Step 3	<code>switch(config-line)# access-class access-list-number {in out}</code> Example: <code>switch(config-line)# access-class ozi2 in</code> <code>switch(config-line)#access-class ozi3 out</code> <code>switch(config)#</code>	Specifies inbound or outbound access restrictions.

	Command or Action	Purpose
Step 4	(Optional) switch(config-line)# no access-class access-list-number {in out} Example: switch(config-line)# no access-class ozi2 in switch(config-line)# no access-class ozi3 out switch(config)#	Removes inbound or outbound access restrictions.
Step 5	switch(config-line)# exit Example: switch(config-line)# exit switch#	Exits line configuration mode.
Step 6	(Optional) switch# show running-config aclmgr Example: switch# show running-config aclmgr	Displays the running configuration of the ACLs on the switch.
Step 7	(Optional) switch# copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to apply the access-class ozi2 command to the in-direction of the vty line.

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# line vty
switch(config-line)# access-class ozi2 in
switch(config-line)# exit
switch#
```

Verifying ACLs on VTY Lines

To display the ACL configurations on VTY lines, perform one of the following tasks:

Command	Purpose
show running-config aclmgr	Displays the running configuration of the ACLs configured on the switch.
show users	Displays the users that are connected.
show access-lists access-list-name	Display the statistics per entry.

Configuration Examples for ACLs on VTY Lines

The following example shows the connected users on the console line (ttyS0) and the VTY lines (pts/0 and pts/1).

```
switch# show users
NAME      LINE      TIME          IDLE          PID COMMENT
admin    ttyS0     Aug 27 20:45  .           14425 *
admin    pts/0     Aug 27 20:06 00:46      14176 (172.18.217.82) session=ssh
admin    pts/1     Aug 27 20:52  .           14584 (10.55.144.118)
```

The following example shows how to allow vty connections to all IPv4 hosts except 172.18.217.82 and how to deny vty connections to any IPv4 host except 10.55.144.118, 172.18.217.79, 172.18.217.82, 172.18.217.92:

- Applying the ipv6 access-list ozi7 command to the in direction of the VTY line, denies VTY connections to all IPv6 hosts.
- Applying the ipv6 access-list ozip6 command to the out direction of the VTY line, allows VTY connections to all IPv6 hosts.

```
switch# show running-config aclmgr
!Time: Fri Aug 27 22:01:09 2010
version 5.0(2)N1(1)
ip access-list ozi
  10 deny ip 172.18.217.82/32 any
  20 permit ip any any
ip access-list ozi2
  10 permit ip 10.55.144.118/32 any
  20 permit ip 172.18.217.79/32 any
  30 permit ip 172.18.217.82/32 any
  40 permit ip 172.18.217.92/32 any
ipv6 access-list ozi7
  10 deny tcp any any
ipv6 access-list ozip6
  10 permit tcp any any

line vty
  access-class ozi in
  access-class ozi2 out
  ipv6 access-class ozi7 in
  ipv6 access-class ozip6 out
```

The following example shows how to configure the IP access list by enabling per-entry statistics for the ACL:

```
switch# conf t
Enter configuration commands, one per line.
End with CNTL/Z.
switch(config)# ip access-list ozi2
switch(config-acl)# statistics per-entry
switch(config-acl)# deny tcp 172.18.217.83/32 any
switch(config-acl)# exit

switch(config)# ip access-list ozi
switch(config-acl)# statistics per-entry
switch(config-acl)# permit ip 172.18.217.20/24 any
switch(config-acl)# exit
switch#
```

The following example shows how to apply the ACLs on VTY in and out directions:

```
switch(config)# line vty
switch(config-line)# ip access-class ozi in
switch(config-line)# access-class ozi2 out
```

```
switch(config-line)# exit
switch#
```

The following example shows how to remove the access restrictions on the VTY line:

```
switch# conf t
Enter configuration commands, one per line. End
with CNTL/Z.
switch(config)# line vty
switch(config-line)# no access-class ozi2 in
switch(config-line)# no ip access-class ozi2 in
switch(config-line)# exit
switch#
```



CHAPTER 10

Configuring Port Security

This chapter includes the following sections:

- [Information About Port Security](#), on page 161
- [Licensing Requirements for Port Security](#), on page 166
- [Prerequisites for Port Security](#), on page 166
- [Guidelines and Limitations for Port Security](#), on page 166
- [Guidelines and Limitations for Port Security on vPCs](#), on page 166
- [Configuring Port Security](#), on page 167
- [Verifying the Port Security Configuration](#), on page 176
- [Displaying Secure MAC Addresses](#), on page 177
- [Configuration Example for Port Security](#), on page 177
- [Configuration Example of Port Security in a vPC Domain](#), on page 177
- [Default Settings for Port Security](#), on page 178
- [Additional References for Port Security](#), on page 178
- [Feature History for Port Security](#), on page 179

Information About Port Security

Port security allows you to configure Layer 2 physical interfaces, Layer 2 port-channel interfaces, and virtual port channels (vPCs) to allow inbound traffic from only a restricted set of MAC addresses. The MAC addresses in the restricted set are called secure MAC addresses. In addition, the device does not allow traffic from these MAC addresses on another interface within the same VLAN. The number of MAC addresses that the device can secure is configurable per interface.



Note Unless otherwise specified, the term *interface* refers to physical interfaces, port-channel interfaces, and vPCs; likewise, the term *Layer 2 interface* refers to both Layer 2 physical interfaces and Layer 2 port-channel interfaces.

Secure MAC Address Learning

The process of securing a MAC address is called learning. A MAC address can be a secure MAC address on one interface only. For each interface that you enable port security on, the device can learn a limited number

of MAC addresses by the static, dynamic, or sticky methods. The way that the device stores secure MAC addresses varies depending upon how the device learned the secure MAC address.



Note All learned MAC addresses are synchronized between vPC peers.

Static Method

The static learning method allows you to manually add or remove secure MAC addresses to the running configuration of an interface. If you copy the running configuration to the startup configuration, static secure MAC addresses are unaffected if the device restarts.

A static secure MAC address entry remains in the configuration of an interface until one of the following events occurs:

- You explicitly remove the address from the configuration.
- You configure the interface to act as a Layer 3 interface.

Adding secure addresses by the static method is not affected by whether dynamic or sticky address learning is enabled.

Dynamic Method

By default, when you enable port security on an interface, you enable the dynamic learning method. With this method, the device secures MAC addresses as ingress traffic passes through the interface. If the address is not yet secured and the device has not reached any applicable maximum, it secures the address and allows the traffic.

The device stores dynamic secure MAC addresses in memory. A dynamic secure MAC address entry remains in the configuration of an interface until one of the following events occurs:

- The device restarts.
- The interface restarts.
- The address reaches the age limit that you configured for the interface.
- You explicitly remove the address.
- You configure the interface to act as a Layer 3 interface.

Sticky Method

If you enable the sticky method, the device secures MAC addresses in the same manner as dynamic address learning, but the device stores addresses learned by this method in nonvolatile RAM (NVRAM). As a result, addresses learned by the sticky method persist through a device restart. Sticky secure MAC addresses do not appear in the running configuration of an interface.

Dynamic and sticky address learning are mutually exclusive. When you enable sticky learning on an interface, the device stops dynamic learning and performs sticky learning instead. If you disable sticky learning, the device resumes dynamic learning.

A sticky secure MAC address entry remains in the configuration of an interface until one of the following events occurs:

- You explicitly remove the address.
- You configure the interface to act as a Layer 3 interface.

Dynamic Address Aging

The device ages MAC addresses learned by the dynamic method and drops them after the age limit is reached. You can configure the age limit on each interface. The range is from 0 to 1440 minutes, where 0 disables aging.

In vPC domains, dynamic MAC addresses are dropped only after the age limit is reached on both vPC peers.

The method that the device uses to determine that the MAC address age is also configurable. The two methods of determining address age are as follows:

Inactivity

The length of time after the device last received a packet from the address on the applicable interface.

Absolute

The length of time after the device learned the address. This is the default aging method; however, the default aging time is 0 minutes, which disables aging.

Secure MAC Address Maximums

By default, an interface can have only one secure MAC address. You can configure the maximum number of MAC addresses permitted per interface or per VLAN on an interface. Maximums apply to secure MAC addresses learned by any method: dynamic, sticky, or static.



Note In vPC domains, the configuration on the primary vPC takes effect.



Tip To ensure that an attached device has the full bandwidth of the port, set the maximum number of addresses to one and configure the MAC address of the attached device.

The following three limits can determine how many secure MAC addresses are permitted on an interface:

Device maximum

The device has a nonconfigurable limit of 8192 secure MAC addresses. If learning a new address would violate the device maximum, the device does not permit the new address to be learned, even if the interface or VLAN maximum has not been reached.

Interface maximum

You can configure a maximum number of 1025 secure MAC addresses for each interface protected by port security. The default interface maximum is one address. Interface maximums cannot exceed the device maximum.

In vPC domains, you set the maximum number of secure MAC addresses on the primary vPC switch. The primary vPC switch does the count validation, even if a maximum number of secure MAC addresses is set on the secondary switch.

VLAN maximum

You can configure the maximum number of secure MAC addresses per VLAN for each interface protected by port security. A VLAN maximum cannot exceed the configured interface maximum. VLAN maximums are useful only for trunk ports. There are no default VLAN maximums.

You can configure VLAN and interface maximums per interface, as needed; however, when the new limit is less than the applicable number of secure addresses, you must reduce the number of secure MAC addresses first.

Security Violations and Actions

Port security triggers security violations when either of the two following events occur:

MAX Count Violation

Ingress traffic arrives at an interface from a nonsecure MAC address and learning the address would exceed the applicable maximum number of secure MAC addresses. The blocked entry is added to the Forwarding Module (FWM) of the Cisco Nexus switch.

When an interface has both a VLAN maximum and an interface maximum configured, a violation occurs when either maximum is exceeded. For example, consider the following on a single interface configured with port security:

- VLAN 1 has a maximum of 5 addresses
- The interface has a maximum of 10 addresses

The device detects a violation when any of the following occurs:

- The device has learned five addresses for VLAN 1 and inbound traffic from a sixth address arrives at the interface in VLAN 1.
- The device has learned 10 addresses on the interface and inbound traffic from an 11th address arrives at the interface.

MAC Move Violation

Ingress traffic from a secure MAC address arrives at a different interface in the same VLAN as the interface on which the address is secured. The blocked entry is added as a drop entry in the Port Security table.

When a security violation occurs, the device increments the security violation counter for the interface and takes the action specified by the port security configuration of the interface. If a violation occurs because ingress traffic from a secure MAC address arrives at a different interface than the interface on which the address is secure, the device applies the action on the interface that received the traffic.

The possible actions that the device can take are as follows:

Shutdown

Shuts down the interface that received the packet triggering the violation. The interface is error disabled. This action is the default. After you reenables the interface, it retains its port security configuration, including its secure MAC addresses.

You can use the **errdisable** global configuration command to configure the device to reenables the interface automatically if a shutdown occurs, or you can manually reenables the interface by entering the **shutdown** and **no shut down** interface configuration commands.

The MAC address does not move to the unsecure port, and the frame on the unsecure port is dropped.

Restrict

Drops ingress traffic from any nonsecure MAC addresses and adds the MAC address as a blocked MAC entry in the port security table..



Note In vPC domains, blocked MAC addresses added to the port security table due to violations occurring in the Restrict mode are not synchronized across vPC peers.

The device keeps a count of the number of dropped packets, which is called the security violation count. Address learning continues until the maximum security violations have occurred on the interface. Traffic from addresses learned after the first security violation is dropped.

After the maximum number of MAX count violations (10) is reached, a violation is triggered and the device stops learning new MAC addresses.

Protect

Prevents further violations from occurring. The address that triggered the security violation is learned but any traffic from the address is dropped. Further address learning stops.



Note In vPCs, the violation action configured on the primary vPC switch takes affect. So, whenever a security violation is triggered, the security action defined on the primary vPC switch occurs.

After the maximum number of MAX move violations (10) is reached, the interface is shut down and placed in the **errdisabled** state.

Port Type Changes

When you have configured port security on a Layer 2 interface and you change the port type of the interface, the device behaves as follows:

Access port to trunk port

When you change a Layer 2 interface from an access port to a trunk port, the device drops all secure addresses learned by the dynamic method. The device moves the addresses learned by the static or sticky method to the native trunk VLAN.

Trunk port to access port

When you change a Layer 2 interface from an access port to a trunk port, the device drops all secure addresses learned by the dynamic method. The device moves the addresses learned by the static method to the native trunk VLAN. The sticky MAC addresses remain in same VLAN if the VLAN exists. Otherwise, the MAC addresses move to the native VLAN of the trunk port.

Switched port to routed port

When you change an interface from a Layer 2 interface to a Layer 3 interface, the device disables port security on the interface and discards all port security configuration for the interface. The device also discards all secure MAC addresses for the interface, regardless of the method used to learn the address.

Routed port to switched port

When you change an interface from a Layer 3 interface to a Layer 2 interface, the device has no port security configuration for the interface.

Licensing Requirements for Port Security

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	Port security requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS device images and is provided at no extra charge to you. For an explanation of the Cisco NX-OS licensing scheme, see the <i>License and Copyright Information for Cisco NX-OS Software</i> available at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/license_agreement/nx-ossw_lisns.htm

Prerequisites for Port Security

Port security has the following prerequisites:

- You must globally enable port security for the device that you want to protect with port security.
- In a vPC domain, you must enable port security globally on both vPC peers and on both vPC interfaces on the vPC peers. We recommend that you use the **config sync** command to ensure that the configuration is consistent on both vPC peers.

Guidelines and Limitations for Port Security

When configuring port security, follow these guidelines:

- Port security is supported on PVLAN ports.
- Port security does not support switched port analyzer (SPAN) destination ports.
- Port security does not depend upon other features.
- Port security is not supported on vPC peer links.
- Port security is not supported on Network Interface (NIF) port, Flex Link ports, or vEthernet interfaces.

Guidelines and Limitations for Port Security on vPCs

In addition to the guidelines and limitations for port security, there are additional guidelines and limitations for port security on vPCs. When configuring port security on vPCs, follow these guidelines:

- You must enable port security globally on both vPC peers in a vPC domain.
- You must enable port security on the vPC interfaces of both vPC peers.

- You must configure a static secure MAC address on the primary vPC peer. This MAC address is synchronized with the secondary vPC peer. Do not configure a static secure MAC address on the secondary peer. This MAC address appears in the secondary vPC configuration, but does not take affect.
- All learned MAC addresses are synchronized between vPC peers.
- Both vPC peers can be configured with either the dynamic or sticky MAC address learning method. However, we recommend that both vPC peers be configured for the same method.
- Dynamic MAC addresses are dropped only after the age limit is reached on both vPC peers.
- You set the maximum number of secure MAC addresses on the primary vPC switch. The primary vPC switch does the count validation, even if a maximum number of secure MAC addresses is set on the secondary switch.
- You configure the violation action on the primary vPC. So, whenever a security violation is triggered, the security action defined on the primary vPC switch occurs.
- Port security is enabled on a vPC interface when the port security feature is enabled on both vPC peers and port security is enabled on both vPC interfaces of the vPC peers. You can use the **config sync** command to verify that the configuration is correct.
- While a switch undergoes an in-service software upgrade (ISSU), port security operations are stopped on its peer switch. The peer switch does not learn any new MAC addresses, and MAC moves occurring during this operation are ignored. When the ISSU is complete, the peer switch is notified and normal port security functionality resumes.
- ISSU to higher versions is supported; however ISSU to lower versions is not supported.

Configuring Port Security

Enabling or Disabling Port Security Globally

You can enable or disable port security globally on a device. By default, port security is disabled globally.

When you disable port security, all port security configuration on the interface is ineffective. When you disable port security globally, all port security configuration is lost.



Note To enable or disable port security in a vPC domain, you must enable or disable port security globally on both vPC peers.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	[no] feature port-security Example: switch(config)# feature port-security	Enables port security globally. The no option disables port security globally.
Step 3	show port-security Example: switch(config)# show port-security	Displays the status of port security.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.
Step 5	If you are configuring port security for a vPC domain, repeat steps 1 through 4 on the vPC peer to enable port security globally. Example:	—

Enabling or Disabling Port Security on a Layer 2 Interface

You can enable or disable port security on a Layer 2 interface. By default, port security is disabled on all interfaces.

When you disable port security on an interface, all switchport port security configuration for the interface is lost.

Before you begin

You must have enabled port security globally.

If you are setting up port security in a vPC domain, you must have enabled port security globally on both vPC peers.

If a Layer 2 Ethernet interface is a member of a port-channel interface, you cannot enable or disable port security on the Layer 2 Ethernet interface.

If any member port of a secure Layer 2 port-channel interface has port security enabled, you cannot disable port security for the port-channel interface unless you first remove all secure member ports from the port-channel interface.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>slot/port</i> • interface port-channel <i>channel-number</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Enters interface configuration mode for the Ethernet or port-channel interface that you want to configure with port security.
Step 3	switchport Example: <pre>switch(config-if)# switchport</pre>	Configures the interface as a Layer 2 interface.
Step 4	[no] switchport port-security Example: <pre>switch(config-if)# switchport port-security</pre>	Enables port security on the interface. The no option disables port security on the interface.
Step 5	show running-config port-security Example: <pre>switch(config-if)# show running-config port-security</pre>	Displays the port security configuration.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.
Step 7	If you are configuring port security for a vPC domain, repeat steps 1 through 6 to on the vPC peer to enable port security on its vPC interface.	—

Enabling or Disabling Sticky MAC Address Learning

You can disable or enable sticky MAC address learning on an interface. If you disable sticky learning, the device returns to dynamic MAC address learning on the interface, which is the default learning method.

By default, sticky MAC address learning is disabled.

Before you begin

You must have enabled port security globally.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>slot/port</i> • interface port-channel <i>channel-number</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Enters interface configuration mode for the interface that you want to configure with sticky MAC address learning.
Step 3	switchport Example: <pre>switch(config-if)# switchport</pre>	Configures the interface as a Layer 2 interface.
Step 4	[no] switchport port-security mac-address sticky Example: <pre>switch(config-if)# switchport port-security mac-address sticky</pre>	Enables sticky MAC address learning on the interface. The no option disables sticky MAC address learning.
Step 5	show running-config port-security Example: <pre>switch(config-if)# show running-config port-security</pre>	Displays the port security configuration.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Adding a Static Secure MAC Address on an Interface

You can add a static secure MAC address on a Layer 2 interface.

**Note**

If the MAC address is a secure MAC address on any interface, you cannot add it as a static secure MAC address to another interface until you remove it from the interface on which it is already a secure MAC address.

By default, no static secure MAC addresses are configured on an interface.

Before you begin

You must have enabled port security globally.

Verify that the interface maximum has not been reached for secure MAC addresses. If needed, you can remove a secure MAC address or you can change the maximum number of addresses on the interface.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>slot/port</i> • interface port-channel <i>channel-number</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Enters interface configuration mode for the interface that you specify.
Step 3	[no] switchport port-security mac-address <i>address [vlan vlan-ID]</i> Example: <pre>switch(config-if)# switchport port-security mac-address 0019.D2D0.00AE</pre>	Configures a static MAC address for port security on the current interface. Use the vlan keyword if you want to specify the VLAN that traffic from the address is allowed on.
Step 4	show running-config port-security Example: <pre>switch(config-if)# show running-config port-security</pre>	Displays the port security configuration.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Removing a Static Secure MAC Address on an Interface

You can remove a static secure MAC address on a Layer 2 interface.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>slot/port</i> • interface port-channel <i>channel-number</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Enters interface configuration mode for the interface from which you want to remove a static secure MAC address.
Step 3	no switchport port-security mac-address <i>address</i> Example: <pre>switch(config-if)# no switchport port-security mac-address 0019.D2D0.00AE</pre>	Removes the static secure MAC address from port security on the current interface.
Step 4	show running-config port-security Example: <pre>switch(config-if)# show running-config port-security</pre>	Displays the port security configuration.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Removing a Dynamic Secure MAC Address

You can remove dynamically learned, secure MAC addresses.

Before you begin

You must have enabled port security globally.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<p>clear port-security dynamic {interface ethernet slot/port address address} [vlan vlan-ID]</p> <p>Example:</p> <pre>switch(config)# clear port-security dynamic interface ethernet 2/1</pre>	<p>Removes dynamically learned, secure MAC addresses, as specified.</p> <p>If you use the interface keyword, you remove all dynamically learned addresses on the interface that you specify.</p> <p>If you use the address keyword, you remove the single, dynamically learned address that you specify.</p> <p>Use the vlan keyword if you want to further limit the command to removing an address or addresses on a particular VLAN.</p>
Step 3	<p>show port-security address</p> <p>Example:</p> <pre>switch(config)# show port-security address</pre>	Displays secure MAC addresses.
Step 4	<p>(Optional) copy running-config startup-config</p> <p>Example:</p> <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring a Maximum Number of MAC Addresses

You can configure the maximum number of MAC addresses that can be learned or statically configured on a Layer 2 interface. You can also configure a maximum number of MAC addresses per VLAN on a Layer 2 interface. The largest maximum number of addresses that you can configure on an interface is 1025 addresses. The system maximum number of address is 8192.

By default, an interface has a maximum of one secure MAC address. VLANs have no default maximum number of secure MAC addresses.



Note When you specify a maximum number of addresses that is less than the number of addresses already learned or statically configured on the interface, the device rejects the command. To remove all addresses learned by the dynamic method, use the **shutdown** and **no shutdown** commands to restart the interface.

Before you begin

You must have enabled port security globally.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>slot/port</i> • interface port-channel <i>channel-number</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Enters interface configuration mode, where <i>slot</i> is the interface that you want to configure with the maximum number of MAC addresses.
Step 3	[no] switchport port-security maximum <i>number</i> [vlan <i>vlan-ID</i>] Example: <pre>switch(config-if)# switchport port-security maximum 425</pre>	Configures the maximum number of MAC addresses that can be learned or statically configured for the current interface. The highest valid <i>number</i> is 1025. The no option resets the maximum number of MAC addresses to the default, which is 1. If you want to specify the VLAN that the maximum applies to, use the vlan keyword.
Step 4	show running-config port-security Example: <pre>switch(config-if)# show running-config port-security</pre>	Displays the port security configuration.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring an Address Aging Type and Time

You can configure the MAC address aging type and the length of time that the device uses to determine when MAC addresses learned by the dynamic method have reached their age limit.

Absolute aging is the default aging type.

By default, the aging time is 0 minutes, which disables aging.

Before you begin

You must have enabled port security globally.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>slot/port</i> • interface port-channel <i>channel-number</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Enters interface configuration mode for the interface that you want to configure with the MAC aging type and time.
Step 3	[no] switchport port-security aging type {absolute inactivity} Example: switch(config-if)# switchport port-security aging type inactivity	Configures the type of aging that the device applies to dynamically learned MAC addresses. The no option resets the aging type to the default, which is absolute aging.
Step 4	[no] switchport port-security aging time <i>minutes</i> Example: switch(config-if)# switchport port-security aging time 120	Configures the number of minutes that a dynamically learned MAC address must age before the device drops the address. The maximum valid <i>minutes</i> is 1440. The no option resets the aging time to the default, which is 0 minutes (no aging).
Step 5	show running-config port-security Example: switch(config-if)# show running-config port-security	Displays the port security configuration.
Step 6	(Optional) copy running-config startup-config Example: switch(config-if)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring a Security Violation Action

You can configure the action that the device takes if a security violation occurs. The violation action is configurable on each interface that you enable with port security.

The default security action is to shut down the port on which the security violation occurs.

Before you begin

You must have enabled port security globally.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>slot/port</i> • interface port-channel <i>channel-number</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Enters interface configuration mode for the interface that you want to configure with a security violation action.
Step 3	[no] switchport port-security violation {protect restrict shutdown} Example: <pre>switch(config-if)# switchport port-security violation restrict</pre>	Configures the security violation action for port security on the current interface. The no option resets the violation action to the default, which is to shut down the interface.
Step 4	show running-config port-security Example: <pre>switch(config-if)# show running-config port-security</pre>	Displays the port security configuration.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying the Port Security Configuration

To display the port security configuration information, perform one of the following tasks. For detailed information about the fields in the output from this command, see the Security command reference for your platform.

Command	Purpose
show running-config port-security	Displays the port security configuration.
show port-security	Displays the port security status of the device.
show port-security interface	Displays the port security status of a specific interface.
show port-security address	Displays secure MAC addresses.

Command	Purpose
<code>show running-config interface</code>	Displays the interfaces that are in the running-configuration.
<code>show mac address-table</code>	Displays the contents of the MAC address table.
<code>show system internal port-security info global</code>	Displays the port security settings of the device.

Displaying Secure MAC Addresses

Use the `show port-security address` command to display secure MAC addresses. For detailed information about the fields in the output from this command, see the *Security command reference for your platform*.

Configuration Example for Port Security

The following example shows a port security configuration for the Ethernet 2/1 interface with VLAN and interface maximums for secure addresses. In this example, the interface is a trunk port. Additionally, the violation action is set to Restrict.

```
feature port-security
interface Ethernet 2/1
  switchport
  switchport port-security
  switchport port-security maximum 10
  switchport port-security maximum 7 vlan 10
  switchport port-security maximum 3 vlan 20
  switchport port-security violation restrict
```

Configuration Example of Port Security in a vPC Domain

The following example shows how to enable and configure port security on vPC peers in a vPC domain. The first switch is the primary vPC peer and the second switch is the secondary vPC peer. It is assumed that domain 103 has already been created.

```
primary_switch(config)# feature port-security
primary_switch(config-if)# int e1/1
primary_switch(config-if)# switchport port-security
primary_switch(config-if)# switchport port-security max 1025
primary_switch(config-if)# switchport port-security violation restrict
primary_switch(config-if)# switchport port-security aging time 4
primary_switch(config-if)# switchport port-security aging type absolute
primary_switch(config-if)# switchport port-security mac sticky
primary_switch(config-if)# switchport port-security mac-address 0.0.1 vlan 101
primary_switch(config-if)# switchport port-security mac-address 0.0.2 vlan 101
primary_switch(config-if)# copy running-config startup-config

secondary_switch(config)# int e103/1/1
secondary_switch(config-if)# switchport port-security
secondary_switch(config-if)# copy running-config startup-config
```

Default Settings for Port Security

This table lists the default settings for port security parameters.

Table 17: Default Port Security Parameters

Parameters	Default
Port security enablement globally	Disabled
Port security enablement per interface	Disabled
MAC address learning method	Dynamic
Interface maximum number of secure MAC addresses	1
Security violation action	Shutdown

Additional References for Port Security

Related Documents

Related Topic	Document Title
Layer 2 switching	<i>Cisco Nexus 5500 Series NX-OS Layer 2 Switching Configuration Guide</i>
Port security commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	<i>Cisco Nexus 5500 Series NX-OS Security Command Reference</i>

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

Cisco NX-OS provides read-only SNMP support for port security.

MIBs	MIBs Link
<ul style="list-style-type: none"> • CISCO-PORT-SECURITY-MIB <p>Note Traps are supported for notification of secure MAC address violations.</p>	To locate and download MIBs, go to the following URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Feature History for Port Security

This table lists the release history for this feature.

Table 18: Feature History for Port Security

Feature Name	Releases	Feature Information
Port security	5.1(3)N1(1)	Feature introduced in this release.



CHAPTER 11

Configuring DHCP Snooping

This chapter contains the following sections:

- [Information About DHCP Snooping, on page 181](#)
- [Information About the DHCP Relay Agent, on page 186](#)
- [Guidelines and Limitations for DHCP Snooping, on page 187](#)
- [Default Settings for DHCP Snooping, on page 188](#)
- [Configuring DHCP Snooping, on page 188](#)
- [Verifying the DHCP Snooping Configuration, on page 198](#)
- [Displaying DHCP Bindings, on page 198](#)
- [Clearing the DHCP Snooping Binding Database, on page 199](#)
- [Configuration Examples for DHCP Snooping, on page 200](#)

Information About DHCP Snooping

DHCP snooping acts like a firewall between untrusted hosts and trusted DHCP servers. DHCP snooping performs the following activities:

- Validates DHCP messages received from untrusted sources and filters out invalid messages.
- Builds and maintains the DHCP snooping binding database, which contains information about untrusted hosts with leased IP addresses.
- Uses the DHCP snooping binding database to validate subsequent requests from untrusted hosts.

DHCP snooping is enabled on a per-VLAN basis. By default, the feature is inactive on all VLANs. You can enable the feature on a single VLAN or a range of VLANs.

Feature Enabled and Globally Enabled

When you are configuring DHCP snooping, it is important that you understand the difference between enabling the DHCP snooping feature and globally enabling DHCP snooping.

Feature Enablement

The DHCP snooping feature is disabled by default. When the DHCP snooping feature is disabled, you cannot configure it or any of the features that depend on DHCP snooping. The commands to configure DHCP snooping and its dependent features are unavailable when DHCP snooping is disabled.

When you enable the DHCP snooping feature, the switch begins building and maintaining the DHCP snooping binding database. Features dependent on the DHCP snooping binding database can now make use of it and can therefore also be configured.

Enabling the DHCP snooping feature does not globally enable it. You must separately enable DHCP snooping globally.

Disabling the DHCP snooping feature removes all DHCP snooping configuration from the switch. If you want to disable DHCP snooping and preserve the configuration, globally disable DHCP snooping but do not disable the DHCP snooping feature.

Global Enablement

After DHCP snooping is enabled, DHCP snooping is globally disabled by default. Global enablement is a second level of enablement that allows you to have separate control of whether the switch is actively performing DHCP snooping that is independent from enabling the DHCP snooping binding database.

When you globally enable DHCP snooping, on each untrusted interface of VLANs that have DHCP snooping enabled, the switch begins validating DHCP messages that are received and used the DHCP snooping binding database to validate subsequent requests from untrusted hosts.

When you globally disable DHCP snooping, the switch stops validating DHCP messages and validating subsequent requests from untrusted hosts. It also removes the DHCP snooping binding database. Globally disabling DHCP snooping does not remove any DHCP snooping configuration or the configuration of other features that are dependent upon the DHCP snooping feature.

Trusted and Untrusted Sources

You can configure whether DHCP snooping trusts traffic sources. An untrusted source might initiate traffic attacks or other hostile actions. To prevent such attacks, DHCP snooping filters messages from untrusted sources.

In an enterprise network, a trusted source is a switch that is under your administrative control. These switches include the switches, routers, and servers in the network. Any switch beyond the firewall or outside the network is an untrusted source. Generally, host ports are treated as untrusted sources.

In a service provider environment, any switch that is not in the service provider network is an untrusted source (such as a customer switch). Host ports are untrusted sources.

In a Cisco Nexus device, you indicate that a source is trusted by configuring the trust state of its connecting interface.

The default trust state of all interfaces is untrusted. You must configure DHCP server interfaces as trusted. You can also configure other interfaces as trusted if they connect to switches (such as switches or routers) inside your network. You usually do not configure host port interfaces as trusted.



Note

For DHCP snooping to function properly, you must connect all DHCP servers to the switch through trusted interfaces.

DHCP Snooping Binding Database

Using information extracted from intercepted DHCP messages, DHCP snooping dynamically builds and maintains a database. The database contains an entry for each untrusted host with a leased IP address if the host is associated with a VLAN that has DHCP snooping enabled. The database does not contain entries for hosts that are connected through trusted interfaces.



Note The DHCP snooping binding database is also referred to as the DHCP snooping binding table.

DHCP snooping updates the database when the switch receives specific DHCP messages. For example, the feature adds an entry to the database when the switch receives a DHCPACK message from the server. The feature removes the entry in the database when the IP address lease expires or the switch receives a DHCPRELEASE message from the host.

Each entry in the DHCP snooping binding database includes the MAC address of the host, the leased IP address, the lease time, the binding type, and the VLAN number and interface information associated with the host.

You can remove entries from the binding database by using the **clear ip dhcp snooping binding** command.

DHCP Snooping Option 82 Data Insertion

DHCP can centrally manage the IP address assignments for a large number of subscribers. When you enable Option 82, the device identifies a subscriber device that connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can connect to the same port on the access device and are uniquely identified.

When you enable Option 82 on the Cisco NX-OS device, the following sequence of events occurs:

1. The host (DHCP client) generates a DHCP request and broadcasts it on the network.
2. When the Cisco NX-OS device receives the DHCP request, it adds the Option 82 information in the packet. The Option 82 information contains the device MAC address (the remote ID suboption) and the port identifier, vlan-mod-port, from which the packet is received (the circuit ID suboption). For hosts behind the port channel, the circuit ID is filled with the if_index of the port channel.



Note For vPC peer switches, the remote ID suboption contains the vPC switch MAC address, which is unique in both switches. This MAC address is computed with the vPC domain ID. The Option 82 information is inserted at the switch where the DHCP request is first received before it is forwarded to the other vPC peer switch.

3. The device forwards the DHCP request that includes the Option 82 field to the DHCP server.
4. 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. The DHCP server echoes the Option 82 field in the DHCP reply.
5. The DHCP server sends the reply to the Cisco NX-OS device. The Cisco NX-OS device verifies that it originally inserted the Option 82 data by inspecting the remote ID and possibly the circuit ID fields. The

Cisco NX-OS device removes the Option 82 field and forwards the packet to the interface that connects to the DHCP client that sent the DHCP request.

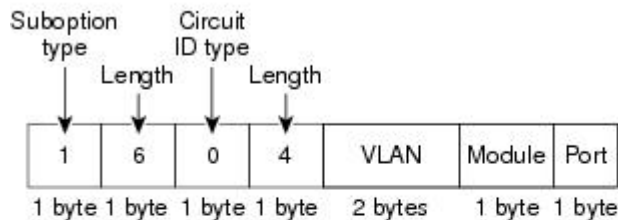
If the previously described sequence of events occurs, the following values do not change:

- 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 circuit ID type

Figure 12: Suboption Packet Formats

This figure shows the packet formats for the remote ID suboption and the circuit ID suboption. The Cisco NX-OS device uses the packet formats when you globally enable DHCP snooping and when you enable Option 82 data insertion and removal. For the circuit ID suboption, the module field is the slot number of the module.

Circuit ID Suboption Frame Format



Remote ID Suboption Frame Format



DHCP Snooping in a vPC Environment

A virtual port channel (vPC) allows two Cisco NX-OS switches to appear as a single logical port channel to a third switch. The third switch can be a switch, server, or any other networking switch that supports port channels.

In a typical vPC environment, DHCP requests can reach one vPC peer switch and the responses can reach the other vPC peer switch, resulting in a partial DHCP (IP-MAC) binding entry in one switch and no binding entry in the other switch. This issue is addressed by using Cisco Fabric Service over Ethernet (CFSoE) distribution to ensure that all DHCP packets (requests and responses) appear on both switches, which helps in creating and maintaining the same binding entry on both switches for all clients behind the vPC link.

CFSoE distribution also allows only one switch to forward the DHCP requests and responses on the vPC link. In non-vPC environments, both switches forward the DHCP packets.

Synchronizing DHCP Snooping Binding Entries

The dynamic DHCP binding entries should be in sync in the following scenarios:

- When the remote vPC is online, all the binding entries for that vPC link should be in sync with the peer.
- When DHCP snooping is enabled on the peer switch, the dynamic binding entries for all vPC links that are up remotely should be in sync with the peer.

Packet Validation

The switch validates DHCP packets received on the untrusted interfaces of VLANs that have DHCP snooping enabled. The switch forwards the DHCP packet unless any of the following conditions occur (in which case, the packet is dropped):

- The switch receives a DHCP response packet (such as a DHCPACK, DHCPNAK, or DHCPOFFER packet) on an untrusted interface.
- The switch receives a packet on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match. This check is performed only if the DHCP snooping MAC address verification option is turned on.
- The switch receives a DHCPRELEASE or DHCPDECLINE message from an untrusted host with an entry in the DHCP snooping binding table, and the interface information in the binding table does not match the interface on which the message was received.
- The switch receives a DHCP packet that includes a relay agent IP address that is not 0.0.0.0.

In addition, you can enable strict validation of DHCP packets, which checks the options field of DHCP packets, including the “magic cookie” value in the first four bytes of the options field. By default, strict validation is disabled. When you enable it, by using the **ip dhcp packet strict-validation** command, if DHCP snooping processes a packet that has an invalid options field, it drops the packet.

Information About the DHCP Relay Agent

DHCP Relay Agent

You can configure the device to run a DHCP relay agent, which forwards DHCP packets between clients and servers. This feature is useful when clients and servers are not on the same physical subnet. Relay agents receive DHCP messages and then generate a new DHCP message to send out on another interface. The relay agent sets the gateway address (giaddr field of the DHCP packet) and, if configured, adds the relay agent information option (Option 82) in the packet and forwards it to the DHCP server. The reply from the server is forwarded back to the client after removing Option 82.

After you enable Option 82, the device uses the binary ifindex format by default. If needed, you can change the Option 82 setting to use an encoded string format instead.

**Note**

When the device relays a DHCP request that already includes Option 82 information, the device forwards the request with the original Option 82 information without altering it.

VRF Support for the DHCP Relay Agent

You can configure the DHCP relay agent to forward DHCP broadcast messages from clients in a virtual routing and forwarding (VRF) instance to DHCP servers in a different VRF. By using a single DHCP server to provide DHCP support to clients in multiple VRFs, you can conserve IP addresses by using a single IP address pool rather than one for each VRF.

Enabling VRF support for the DHCP relay agent requires that you enable Option 82 for the DHCP relay agent.

If a DHCP request arrives on an interface that you have configured with a DHCP relay address and VRF information and the address of the DHCP server belongs to a network on an interface that is a member of a different VRF, the device inserts Option 82 information in the request and forwards it to the DHCP server in the server VRF. The Option 82 information includes the following:

VPN identifier

Name of the VRF that the interface that receives the DHCP request is a member of.

Link selection

Subnet address of the interface that receives the DHCP request.

Server identifier override

IP address of the interface that receives the DHCP request.

**Note**

The DHCP server must support the VPN identifier, link selection, and server identifier override options.

When the device receives the DHCP response message, it strips off the Option 82 information and forwards the response to the DHCP client in the client VRF.

DHCP Relay Binding Database

A relay binding is an entity that associates a DHCP or BOOTP client with a relay agent address and its subnet. Each relay binding stores the client MAC address, active relay agent address, active relay agent address mask, logical and physical interfaces to which the client is connected, giaddr retry count, and total retry count. The giaddr retry count is the number of request packets transmitted with that relay agent address, and the total retry count is the total number of request packets transmitted by the relay agent. One relay binding entry is maintained for each DHCP or BOOTP client.



Note When DHCP smart relay is enabled globally or at the interface level on any switch, the relay bindings on all switches should be synchronized with the vPC peer.

Guidelines and Limitations for DHCP Snooping

Consider the following guidelines and limitations when configuring DHCP snooping:

- The DHCP snooping database can store 2000 bindings.
- DHCP snooping is not active until you enable the feature, enable DHCP snooping globally, and enable DHCP snooping on at least one VLAN.
- Before globally enabling DHCP snooping on the switch, make sure that the switches that act as the DHCP server and the DHCP relay agent are configured and enabled.
- If a VLAN ACL (VACL) is configured on a VLAN that you are configuring with DHCP snooping, ensure that the VACL permits DHCP traffic between DHCP servers and DHCP hosts.
- By default, DHCP bindings are not saved persistently across switch reboots. To maintain persistent bindings across switch reboots, use the **copy r s** command. When the **copy r s** command is issued, all bindings that exist at that time are made persistent across switch reboots.
- Make sure that the DHCP configuration is synchronized across the switches in a vPC link. Otherwise, a run-time error can occur, resulting in dropped packets.
- To use both remote and local DHCP servers, you must configure the DHCP relay feature and either define the unicast address of the local DHCP server or configure a local broadcast address for the subnet where the local DHCP server resides. If you do not define the unicast address of the DHCP server or configure a local broadcast address for the subnet, local DHCP packets cannot be delivered. For example, this situation can occur when you apply an IP DHCP address to an SVI.

The following additional guidelines and limitations apply to implementations that include FabricPath:

- DHCP snooping should be enabled on CE-Fabric boundary switches.
- DHCP snooping is enabled on all access layer switches to secure the network at the access layer.
- DHCP does not learn which binding entries are on ports configured in FabricPath mode. DHCP snooping must be manually enabled on all access layer switches.
- When Dynamic ARP Inspection (DAI) is enabled, ARP packets received on FabricPath ports are allowed.
- IPSG cannot be enabled on ports in FabricPath mode.

- All FabricPath ports in the system must be configured as trusted ports.
- DHCP snooping with Fabric Path has to be enabled on all of the configured VLANs for a switch. If you do not enable FabricPath for all of the VLANs on the switch, DHCP packets will drop for the VLANs where DHCP has not been enabled.

To ensure that DHCP packets are not dropped, you must complete all of the following configurations:

- Enable the DHCP feature using the **feature dhcp** command.
- Install the FabricPath feature set using the **install feature-set fabricpath** and **feature-set fabricpath** commands
- Globally enable DHCP snooping using the **ip dhcp snooping** command.
- Enable DHCP snooping for each of the configured VLANs on the switch using the **ip dhcp snooping vlan *vlan*** command.

Default Settings for DHCP Snooping

This table lists the default settings for DHCP snooping parameters.

Table 19: Default DHCP Snooping Parameters

Parameters	Default
DHCP snooping feature	Disabled
DHCP snooping globally enabled	No
DHCP snooping VLAN	None
DHCP snooping Option 82 support	Disabled
DHCP snooping trust	Untrusted

Configuring DHCP Snooping

Minimum DHCP Snooping Configuration

Procedure

	Command or Action	Purpose
Step 1	Enable the DHCP snooping feature.	When the DHCP snooping feature is disabled, you cannot configure DHCP snooping. For details, see Enabling or Disabling the DHCP Snooping Feature , on page 189.

	Command or Action	Purpose
Step 2	Enable DHCP snooping globally.	For details, see Enabling or Disabling DHCP Snooping Globally , on page 190.
Step 3	Enable DHCP snooping on at least one VLAN.	By default, DHCP snooping is disabled on all VLANs. For details, see Enabling or Disabling DHCP Snooping on a VLAN , on page 190.
Step 4	Ensure that the DHCP server is connected to the switch using a trusted interface.	For details, see Configuring an Interface as Trusted or Untrusted , on page 192.

Enabling or Disabling the DHCP Snooping Feature

You can enable or disable the DHCP snooping feature on the switch. By default, DHCP snooping is disabled.

Before you begin

If you disable the DHCP snooping feature, all DHCP snooping configuration is lost. If you want to turn off DHCP snooping and preserve the DHCP snooping configuration, disable DHCP globally.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] feature dhcp Example: switch(config)# feature dhcp	Enables the DHCP snooping feature. The no option disables the DHCP snooping feature and erases all DHCP snooping configuration.
Step 3	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Shows the DHCP snooping configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling DHCP Snooping Globally

You can enable or disable the DHCP snooping globally on the switch. Globally disabling DHCP snooping stops the switch from performing any DHCP snooping or relaying DHCP messages but preserves DHCP snooping configuration.

Before you begin

Ensure that you have enabled the DHCP snooping feature. By default, DHCP snooping is globally disabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ip dhcp snooping Example: switch(config)# ip dhcp snooping	Enables DHCP snooping globally. The no option disables DHCP snooping.
Step 3	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Shows the DHCP snooping configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling DHCP Snooping on a VLAN

You can enable or disable DHCP snooping on one or more VLANs.

Before you begin

By default, DHCP snooping is disabled on all VLANs.

Ensure that DHCP snooping is enabled.



Note

If a VACL is configured on a VLAN that you are configuring with DHCP snooping, ensure that the VACL permits DHCP traffic between DHCP servers and DHCP hosts.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ip dhcp snooping vlan <i>vlan-list</i> Example: switch(config)# ip dhcp snooping vlan 100,200,250-252	Enables DHCP snooping on the VLANs specified by <i>vlan-list</i> . The no option disables DHCP snooping on the VLANs specified.
Step 3	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Shows the DHCP snooping configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling Option 82 Data Insertion and Removal

You can enable or disable the insertion and removal of Option 82 information for DHCP packets forwarded without the use of the DHCP relay agent.

Before you begin

By default, the switch does not include Option 82 information in DHCP packets.

Ensure that DHCP snooping is enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ip dhcp snooping information option Example: switch(config)# ip dhcp snooping information option	Enables the insertion and removal of Option 82 information from DHCP packets. The no option disables the insertion and removal of Option 82 information.

	Command or Action	Purpose
Step 3	show running-config dhcp Example: switch(config)# show running-config dhcp	Shows the DHCP snooping configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling Strict DHCP Packet Validation

You can enable or disable the strict validation of DHCP packets by the DHCP snooping feature. By default, strict validation of DHCP packets is disabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ip dhcp packet strict-validation Example: switch(config)# ip dhcp packet strict-validation	Enables the strict validation of DHCP packets by the DHCP snooping feature. The no option disables strict DHCP packet validation.
Step 3	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Shows the DHCP snooping configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring an Interface as Trusted or Untrusted

You can configure whether an interface is a trusted or untrusted source of DHCP messages. You can configure DHCP trust on the following types of interfaces:

- Layer 2 Ethernet interfaces

- Layer 2 port-channel interfaces

Before you begin

By default, all interfaces are untrusted.

Ensure that DHCP snooping is enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • interface ethernet <i>port/slot</i> • interface port-channel <i>channel-number</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	<ul style="list-style-type: none"> • Enters interface configuration mode, where <i>port / slot</i> is the Layer 2 Ethernet interface that you want to configure as trusted or untrusted for DHCP snooping. • Enters interface configuration mode, where <i>port / slot</i> is the Layer 2 port-channel interface that you want to configure as trusted or untrusted for DHCP snooping.
Step 3	[no] ip dhcp snooping trust Example: <pre>switch(config-if)# ip dhcp snooping trust</pre>	Configures the interface as a trusted interface for DHCP snooping. The no option configures the port as an untrusted interface.
Step 4	(Optional) show running-config dhcp Example: <pre>switch(config-if)# show running-config dhcp</pre>	Shows the DHCP snooping configuration.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Enabling or Disabling the DHCP Relay Agent

You can enable or disable the DHCP relay agent. By default, the DHCP relay agent is enabled.

Before you begin

Ensure that the DHCP feature is enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ip dhcp relay Example: switch(config)# ip dhcp relay	Enables the DHCP relay agent. The no option disables the DHCP relay agent.
Step 3	(Optional) show ip dhcp relay Example: switch(config)# show ip dhcp relay	Displays the DHCP relay configuration.
Step 4	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Displays the DHCP configuration.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling Option 82 for the DHCP Relay Agent

You can enable or disable the device to insert and remove Option 82 information on DHCP packets forwarded by the relay agent.

By default, the DHCP relay agent does not include Option 82 information in DHCP packets.

Before you begin

Ensure that the DHCP feature is enabled.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# [no] ip dhcp relay information option	Enables the DHCP relay agent to insert and remove Option 82 information on the packets that it forwards. The Option 82 information is in binary ifindex format by default. The no option disables this behavior.

	Command or Action	Purpose
Step 3	(Optional) switch(config)# ip dhcp relay information sub-option circuit-id format-type string	Configures Option 82 to use encoded string format instead of the default binary ifindex format.
Step 4	(Optional) switch(config)# show ip dhcp relay	Displays the DHCP relay configuration.
Step 5	(Optional) switch(config)# show running-config dhcp	Displays the DHCP configuration.
Step 6	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Enabling or Disabling VRF Support for the DHCP Relay Agent

You can configure the device to support the relaying of DHCP requests that arrive on an interface in one VRF to a DHCP server in a different VRF.

Before you begin

You must enable Option 82 for the DHCP relay agent.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ip dhcp relay information option vpn Example: switch(config)# ip dhcp relay information option vpn	Enables VRF support for the DHCP relay agent. The no option disables this behavior.
Step 3	[no] ip dhcp relay sub-option type cisco Example: switch(config)# ip dhcp relay sub-option type cisco	Enables DHCP to use Cisco proprietary numbers 150, 152, and 151 when filling the link selection, server ID override, and VRF name/VPN ID relay agent Option 82 suboptions. The no option causes DHCP to use RFC numbers 5, 11, and 151 for the link selection, server ID override, and VRF name/VPN ID suboptions.
Step 4	(Optional) show ip dhcp relay Example: switch(config)# show ip dhcp relay	Displays the DHCP relay configuration.

	Command or Action	Purpose
Step 5	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Displays the DHCP configuration.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling Subnet Broadcast Support for the DHCP Relay Agent on a Layer 3 Interface

You can configure the device to support the relaying of DHCP packets from clients to a subnet broadcast IP address. When this feature is enabled, the VLAN ACLs (VACLs) accept IP broadcast packets and all subnet broadcast (primary subnet broadcast as well as secondary subnet broadcast) packets.

Before you begin

Ensure that the DHCP feature is enabled.

Ensure that the DHCP relay agent is enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface interface slot/port Example: switch(config)# interface ethernet 2/2 switch(config-if)#	Enters interface configuration mode, where <i>slot/port</i> is the interface for which you want to enable or disable subnet broadcast support for the DHCP relay agent.
Step 3	[no] ip dhcp relay subnet-broadcast Example: switch(config-if)# ip dhcp relay subnet-broadcast	Enables subnet broadcast support for the DHCP relay agent. The no option disables this behavior.
Step 4	exit Example: switch(config-if)# exit switch(config)#	Exits interface configuration mode.

	Command or Action	Purpose
Step 5	exit Example: switch(config)# exit switch#	Exits global configuration mode.
Step 6	(Optional) show ip dhcp relay Example: switch# show ip dhcp relay	Displays the DHCP relay configuration.
Step 7	(Optional) show running-config dhcp Example: switch# show running-config dhcp	Displays the DHCP configuration.
Step 8	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Creating a DHCP Static Binding

You can create a static DHCP source binding to a Layer 2 interface.

Before you begin

Ensure that you have enabled the DHCP snooping feature.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	ip source binding <i>IP-address MAC-address</i> vlan <i>vlan-id</i> { interface ethernet <i>slot/port</i> port-channel <i>channel-no</i> } Example: switch(config)# ip source binding 10.5.22.7 001f.28bd.0013 vlan 100 interface ethernet 2/3	Binds the static source address to the Layer 2 Ethernet interface.
Step 3	(Optional) show ip dhcp snooping binding Example: switch(config)# ip dhcp snooping binding	Shows the DHCP snooping static and dynamic bindings.

	Command or Action	Purpose
Step 4	(Optional) show ip dhcp snooping binding dynamic Example: switch(config)# ip dhcp snooping binding dynamic	Shows the DHCP snooping dynamic bindings.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to create a static IP source entry associated with VLAN 100 on Ethernet interface 2/3:

```
switch# configure terminal
switch(config)# ip source binding 10.5.22.7 001f.28bd.0013 vlan 100 interface ethernet 2/3
switch(config)#
```

Verifying the DHCP Snooping Configuration

To display DHCP snooping configuration information, perform one of the following tasks. For detailed information about the fields in the output from these commands, see the *System Management Configuration Guide* for your Cisco Nexus device.

Command	Purpose
show running-config dhcp	Displays the DHCP snooping configuration.
show ip dhcp snooping	Displays general information about DHCP snooping.

Displaying DHCP Bindings

Use the **show ip dhcp snooping binding** command to display the DHCP static and dynamic binding table. Use the **show ip dhcp snooping binding dynamic** to display the DHCP dynamic binding table.

For detailed information about the fields in the output from this command, see the *System Management Configuration Guide* for your Cisco Nexus device.

This example shows how to create a static DHCP binding and then verify the binding using the **show ip dhcp snooping binding** command.

```
switch# configuration terminal
switch(config)# ip source binding 10.20.30.40 0000.1111.2222 vlan 400 interface port-channel 500
```



```
switch(config)# show ip dhcp snooping binding
-----
MacAddress      IPAddress      LeaseSec      Type      VLAN      Interface
-----
00:00:11:11:22:22  10.20.30.40    infinite      static    400      port-channel500
```

Clearing the DHCP Snooping Binding Database

You can remove entries from the DHCP snooping binding database, including a single entry, all entries associated with an interface, or all entries in the database.

Before you begin

Ensure that DHCP snooping is enabled.

Procedure

	Command or Action	Purpose
Step 1	(Optional) clear ip dhcp snooping binding Example: switch# clear ip dhcp snooping binding	Clears all entries from the DHCP snooping binding database.
Step 2	(Optional) clear ip dhcp snooping binding interface ethernet <i>slot/port[.subinterface-number]</i> Example: switch# clear ip dhcp snooping binding interface ethernet 1/4	Clears entries associated with a specific Ethernet interface from the DHCP snooping binding database.
Step 3	(Optional) clear ip dhcp snooping binding interface port-channel <i>channel-number[.subchannel-number]</i> Example: switch# clear ip dhcp snooping binding interface port-channel 72	Clears entries associated with a specific port-channel interface from the DHCP snooping binding database.
Step 4	(Optional) clear ip dhcp snooping binding vlan <i>vlan-id</i> mac <i>mac-address</i> ip <i>ip-address</i> interface {ethernet <i>slot/port[.subinterface-number]</i> port-channel <i>channel-number[.subchannel-number]</i> } Example: switch# clear ip dhcp snooping binding vlan 23 mac 0060.3aeb.54f0 ip 10.34.54.9 interface ethernet 2/11	Clears a single, specific entry from the DHCP snooping binding database.

	Command or Action	Purpose
Step 5	(Optional) show ip dhcp snooping binding Example: switch# show ip dhcp snooping binding	Displays the DHCP snooping binding database.

Configuration Examples for DHCP Snooping

The following example shows how to enable DHCP snooping on two VLANs, with Option 82 support enabled and Ethernet interface 2/5 trusted because the DHCP server is connected to that interface:

```
feature dhcp
ip dhcp snooping
ip dhcp snooping info option

interface Ethernet 2/5
 ip dhcp snooping trust
ip dhcp snooping vlan 1
ip dhcp snooping vlan 50
```



CHAPTER 12

Configuring Dynamic ARP Inspection

This chapter contains the following sections:

- [Information About DAI, on page 201](#)
- [Licensing Requirements for DAI, on page 204](#)
- [Prerequisites for DAI, on page 205](#)
- [Guidelines and Limitations for DAI, on page 205](#)
- [Default Settings for DAI, on page 206](#)
- [Configuring DAI, on page 206](#)
- [Verifying the DAI Configuration, on page 211](#)
- [Monitoring and Clearing DAI Statistics, on page 211](#)
- [Configuration Examples for DAI, on page 212](#)

Information About DAI

ARP

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. In ARP terms, host B is the sender and host A is the target.

To get the MAC address of host A, 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.

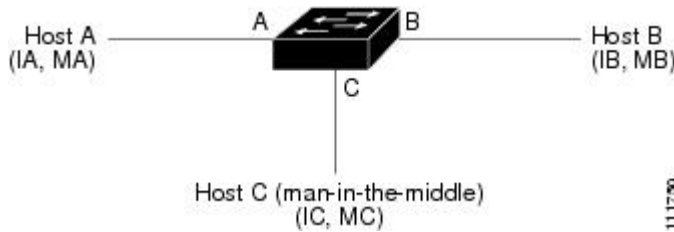
ARP Spoofing Attacks

ARP spoofing attacks and ARP cache poisoning can occur because ARP allows a reply from a host even if an ARP request was not received. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

An ARP spoofing attack can affect hosts, switches, and routers connected to your Layer 2 network by sending false information to the ARP caches of the devices connected to the subnet. Sending false information to an ARP cache is known as ARP cache poisoning. Spoof attacks can also intercept traffic intended for other hosts on the subnet.

Figure 13: ARP Cache Poisoning

This figure shows an example of ARP cache poisoning.



Hosts A, B, and C are connected to the device on interfaces A, B, and C, 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 send IP data to host B, it broadcasts an ARP request for the MAC address associated with IP address IB. When the device 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 device 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 device, host A, and host B by broadcasting two forged ARP responses with bindings: one for a host with an IP address of IA and a MAC address of MC and another for a host with the IP address of IB and a MAC address of MC. Host B and the device then use the MAC address MC as the destination MAC address for traffic intended for IA, which means that host C intercepts that traffic. Likewise, host A and the device use the MAC address MC as the destination MAC address for traffic intended for IB.

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. This topology, in which host C has inserted itself into the traffic stream from host A to host B, is an example of a *man-in-the-middle* attack.

DAI and ARP Spoofing Attacks

DAI ensures that only valid ARP requests and responses are relayed. When DAI is enabled and properly configured, a Cisco Nexus device 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

DAI can determine the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a Dynamic Host Configuration Protocol (DHCP) snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the device. It can also contain static entries that you create. If the ARP packet is received on a trusted interface, the device forwards the packet without any checks. On untrusted interfaces, the device forwards the packet only if it is valid.

You can configure DAI 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.

Related Topics

[Logging DAI Packets](#), on page 204

[Enabling or Disabling Additional Validation](#), on page 208

Interface Trust States and Network Security

DAI associates a trust state with each interface on the device. Packets that arrive on trusted interfaces bypass all DAI validation checks, and packets that arrive on untrusted interfaces go through the DAI validation process.

In a typical network configuration, the guidelines for configuring the trust state of interfaces are as follows:

Untrusted

Interfaces that are connected to hosts

Trusted

Interfaces that are connected to devices

With this configuration, all ARP packets that enter the network from a device bypass the security check. No other validation is needed at any other place in the VLAN or in the network.

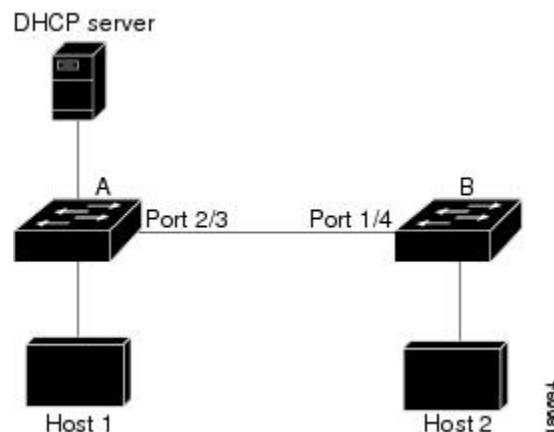


Caution

Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

Figure 14: ARP Packet Validation on a VLAN Enabled for DAI

The following figure, assume that both device A and device B are running DAI 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 device A, only device A binds the IP-to-MAC address of host 1. If the interface between device A and device B is untrusted, the ARP packets from host 1 are dropped by device B and connectivity between host 1 and host 2 is lost.



If you configure interfaces as trusted when they should be untrusted, you may open a security hole in a network. If device A is not running DAI, host 1 can easily poison the ARP cache of device B (and host 2, if you configured the link between the devices as trusted). This condition can occur even though device B is running DAI.

DAI ensures that hosts (on untrusted interfaces) connected to a device that runs DAI do not poison the ARP caches of other hosts in the network; however, DAI does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a device that runs DAI.

If some devices in a VLAN run DAI and other devices do not, the guidelines for configuring the trust state of interfaces on a device that runs DAI becomes the following:

Untrusted

Interfaces that are connected to hosts or to devices that *are not* running DAI

Trusted

Interfaces that are connected to devices that *are* running DAI

To validate the bindings of packets from devices that do not run DAI, configure ARP ACLs on the device that runs DAI. When you cannot determine the bindings, isolate at Layer 3 the devices that run DAI from devices that do not run DAI.



Note Depending on your network setup, you may not be able to validate a given ARP packet on all devices in the VLAN.

Related Topics

[Configuring the DAI Trust State of a Layer 2 Interface](#), on page 207

Logging DAI Packets

Cisco NX-OS maintains a buffer of log entries about DAI packets processed. 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 can also specify the type of packets that are logged. By default, a Cisco Nexus device logs only packets that DAI drops.

If the log buffer overflows, the device overwrites the oldest DAI log entries with newer entries. You can configure the maximum number of entries in the buffer.



Note Cisco NX-OS does not generate system messages about DAI packets that are logged.

Related Topics

[Configuring the DAI Logging Buffer Size](#), on page 209

[Configuring DAI Log Filtering](#), on page 210

Licensing Requirements for DAI

This table shows the licensing requirements for DAI.

Product	License Requirement
Cisco NX-OS	DAI requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For an explanation of the Cisco NX-OS licensing scheme, see the <i>Cisco NX-OS Licensing Guide</i> .

Prerequisites for DAI

- You must enable the DHCP feature before you can configure DAI.

Guidelines and Limitations for DAI

DAI has the following configuration guidelines and limitations:

- DAI is an ingress security feature; it does not perform any egress checking.
- DAI is not effective for hosts connected to devices that do not support DAI or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, you should separate the domain with DAI from domains without DAI. This separation secures the ARP caches of hosts in the domain with DAI.
- DAI depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. If you want DAI to use static IP-MAC address bindings to determine if ARP packets are valid, DHCP snooping needs only to be enabled. If you want DAI to use dynamic IP-MAC address bindings to determine if ARP packets are valid, you must configure DHCP snooping on the same VLANs on which you configure DAI.
- When you use the **feature dhcp** command to enable the DHCP feature, there is a delay of approximately 30 seconds before the I/O modules receive the DHCP or DAI configuration. This delay occurs regardless of the method that you use to change from a configuration with the DHCP feature disabled to a configuration with the DHCP feature enabled. For example, if you use the Rollback feature to revert to a configuration that enables the DHCP feature, the I/O modules receive the DHCP and DAI configuration approximately 30 seconds after you complete the rollback.
- DAI is supported on access ports, trunk ports, port-channel ports, and private VLAN ports.
- The DAI trust configuration of a port channel determines the trust state of all physical ports that you assign to the port channel. For example, if you have configured a physical port as a trusted interface and then you add that physical port to a port channel that is an untrusted interface, the physical port becomes untrusted.
- When you remove a physical port from a port channel, the physical port does not retain the DAI trust state configuration of the port channel.
- When you change the trust state on the port channel, the device configures a new trust state on all the physical ports that comprise the channel.
- If you want DAI to use static IP-MAC address bindings to determine if ARP packets are valid, ensure that DHCP snooping is enabled and that you have configured the static IP-MAC address bindings.
- If you want DAI to use dynamic IP-MAC address bindings to determine if ARP packets are valid, ensure that DHCP snooping is enabled.

Default Settings for DAI

This table lists the default settings for DAI parameters.

Table 20: Default DAI Parameters

Parameters	Default
DAI	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Validation checks	No checks are performed.
Log buffer	When DAI 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.

Configuring DAI

Enabling or Disabling DAI on VLANs

You can enable or disable DAI on VLANs. By default, DAI is disabled on all VLANs.

Before you begin

If you are enabling DAI, ensure the following:

- Ensure that the DHCP feature is enabled.
- The VLANs on which you want to enable DAI are configured.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	[no] ip arp inspection vlan <i>list</i> Example: switch(config)# ip arp inspection vlan 13	Enables DAI for the specified list of VLANs. The no option disables DAI for the specified VLANs.
Step 3	(Optional) show ip arp inspection vlan <i>list</i> Example: switch(config)# show ip arp inspection vlan 13	Shows the DAI status for the specified list of VLANs.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the DAI Trust State of a Layer 2 Interface

You can configure the DAI interface trust state of a Layer 2 interface. By default, all interfaces are untrusted.

A device forwards ARP packets that it receives on a trusted Layer 2 interface but does not check them.

On untrusted interfaces, the device intercepts all ARP requests and responses and verifies that the intercepted packets have valid IP-MAC address bindings before updating the local cache and forwarding the packet to the appropriate destination. If the device determines that packets have invalid bindings, it drops the packets and logs them according to the logging configuration.

Before you begin

If you are enabling DAI, ensure that the DHCP feature is enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface <i>type number / slot</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Enters interface configuration mode.

	Command or Action	Purpose
Step 3	[no] ip arp inspection trust Example: <pre>switch(config-if)# ip arp inspection trust</pre>	Configures the interface as a trusted ARP interface. The no option configures the interface as an untrusted ARP interface.
Step 4	(Optional) show ip arp inspection interface <i>type number / slot</i> Example: <pre>switch(config-if)# show ip arp inspection interface ethernet 2/1</pre>	Displays the trust state and the ARP packet rate for the specified interface.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Related Topics

[Interface Trust States and Network Security](#), on page 203

[Configuring DAI Log Filtering](#), on page 210

Enabling or Disabling Additional Validation

You can enable or disable additional validation of ARP packets. By default, no additional validation of ARP packets is enabled. When no additional validation is configured, the source MAC address and the source IP address check against the IP-to-MAC binding entry for ARP packets are done by using the Ethernet source MAC address (not the ARP sender MAC address) and the ARP sender IP address.

DAI intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can enable additional validation on the destination MAC address, the sender and target IP addresses, and the source MAC address.

You can use the following keywords with the **ip arp inspection validate** command to implement additional validations:

dst-mac

Checks the destination MAC address in the Ethernet header against the target MAC address in the ARP body for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped.

ip

Checks 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.

src-mac

Checks the source MAC address in the Ethernet header against the sender MAC address in the ARP body for ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped.

When enabling additional validation, follow these guidelines:

- You must specify at least one of the keywords. You can specify one, two, or all three keywords.
- Each **ip arp inspection validate** command that you enter replaces the configuration from any previous commands. If you enter an **ip arp inspection validate** command to enable src-mac and dst-mac validations, and a second **ip arp inspection validate** command to enable ip validation, the src-mac and dst-mac validations are disabled when you enter the second command.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ip arp inspection validate {[src-mac] [dst-mac] [ip]} Example: switch(config)# ip arp inspection validate src-mac dst-mac ip	Enables additional DAI validation, or if you use the no option, disables additional DAI validation.
Step 3	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Displays the DHCP snooping configuration, including the DAI configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the DAI Logging Buffer Size

You can configure the DAI logging buffer size. The default buffer size is 32 messages.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	[no] ip arp inspection log-buffer entries number Example: switch(config)# ip arp inspection log-buffer entries 64	Configures the DAI logging buffer size. The no option reverts to the default buffer size, which is 32 messages. The buffer size can be between 0 and 2048 messages.
Step 3	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Displays the DHCP snooping configuration, including the DAI configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring DAI Log Filtering

You can configure how the device determines whether to log a DAI packet. By default, the device logs DAI packets that are dropped.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> ip arp inspection vlan <i>vlan-list</i> logging dhcp-bindings all ip arp inspection vlan <i>vlan-list</i> logging dhcp-bindings none ip arp inspection vlan <i>vlan-list</i> logging dhcp-bindings permit no ip arp inspection vlan <i>vlan-list</i> logging dhcp-bindings {all none permit} Example: switch(config)# ip arp inspection vlan 100 dhcp-bindings permit	Configures DAI log filtering, as follows. The no option removes DAI log filtering. <ul style="list-style-type: none"> Logs all packets that match DHCP bindings. Does not log packets that match DHCP bindings. Logs packets permitted by DHCP bindings. Removes DAI log filtering.

	Command or Action	Purpose
Step 3	(Optional) show running-config dhcp Example: switch(config)# show running-config dhcp	Displays the DHCP snooping configuration, including the DAI configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying the DAI Configuration

To display the DAI configuration information, perform one of the following tasks.

Command	Purpose
show running-config arp	Displays DAI configuration.
show ip arp inspection	Displays the status of DAI.
show ip arp inspection interface ethernet	Displays the trust state and ARP packet rate for a specific interface.
show ip arp inspection vlan	Displays the DAI configuration for a specific VLAN.
show arp access-lists	Displays ARP ACLs.
show ip arp inspection log	Displays the DAI log configuration.

Monitoring and Clearing DAI Statistics

To monitor and clear DAI statistics, use the commands in this table. For more information about these commands, see the *Security Command Reference* for your Cisco Nexus device.

Command	Purpose
show ip arp inspection statistics	Displays DAI statistics.
show ip arp ethernet	Displays interface-specific DAI statistics.
clear ip arp inspection statistics	Clears DAI statistics.

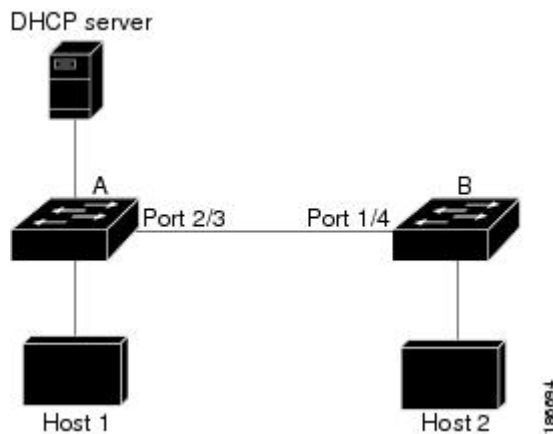
Configuration Examples for DAI

Example 1-Two Devices Support DAI

These procedures show how to configure DAI when two devices support DAI.

Figure 15: Two Devices Supporting DAI

The following figure shows the network configuration for this example. Host 1 is connected to device A, and Host 2 is connected to device B. Both devices are running DAI on VLAN 1 where the hosts are located. A DHCP server is connected to device A. Both hosts acquire their IP addresses from the same DHCP server. Device A has the bindings for Host 1 and Host 2, and device B has the binding for Host 2. Device A Ethernet interface 2/3 is connected to the device B Ethernet interface 1/4.



DAI 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.

- This configuration does not work if the DHCP server is moved from device A to a different location.
- To ensure that this configuration does not compromise security, configure Ethernet interface 2/3 on device A and Ethernet interface 1/4 on device B as trusted.

Configuring Device A

To enable DAI and configure Ethernet interface 2/3 on device A as trusted, follow these steps:

Procedure

Step 1 While logged into device A, verify the connection between device A and device B.

```

switchA# show cdp neighbors
Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  V - VoIP-Phone, D - Remotely-Managed-Device,
                  s - Supports-STP-Dispute
Device ID         Local Intrfce  Hldtme  Capability  Platform  Port ID
  
```

```
switchB          Ethernet2/3    177    R S I    WS-C2960-24TC Ethernet1/4
switchA#
```

Step 2 Enable DAI on VLAN 1 and verify the configuration.

```
switchA# config t
switchA(config)# ip arp inspection vlan 1
switchA(config)# show ip arp inspection vlan 1
Source Mac Validation      : Disabled
Destination Mac Validation : Disabled
IP Address Validation      : Disabled
Vlan : 1
-----
Configuration      : Enabled
Operation State    : Active
switchA(config)#
```

Step 3 Configure Ethernet interface 2/3 as trusted.

```
switchA(config)# interface ethernet 2/3
switchA(config-if)# ip arp inspection trust
switchA(config-if)# exit
switchA(config)# exit
switchA# show ip arp inspection interface ethernet 2/3
Interface      Trust State    Rate (pps)    Burst Interval
-----
Ethernet2/3    Trusted        15            5
```

Step 4 Verify the bindings.

```
switchA# show ip dhcp snooping binding
MacAddress      IpAddress      LeaseSec    Type          VLAN    Interface
-----
00:60:0b:00:12:89  10.0.0.1      0           dhcp-snooping  1      Ethernet2/3
switchA#
```

Step 5 Check the statistics before and after DAI processes any packets.

```
switchA# show ip arp inspection statistics vlan 1
Vlan : 1
-----
ARP Req Forwarded = 0
ARP Res Forwarded = 0
ARP Req Dropped   = 0
ARP Res Dropped   = 0
DHCP Drops        = 0
DHCP Permits      = 0
SMAC Fails-ARP Req = 0
SMAC Fails-ARP Res = 0
DMAC Fails-ARP Res = 0
IP Fails-ARP Req  = 0
IP Fails-ARP Res  = 0
switchA#
```

If host 1 sends out two ARP requests with an IP address of 10.0.0.1 and a MAC address of 0002.0002.0002, both requests are permitted, and are shown as follows:

```
switchA# show ip arp inspection statistics vlan 1
Vlan : 1
-----
```

```

ARP Req Forwarded = 2
ARP Res Forwarded = 0
ARP Req Dropped   = 0
ARP Res Dropped   = 0
DHCP Drops        = 0
DHCP Permits      = 2
SMAC Fails-ARP Req = 0
SMAC Fails-ARP Res = 0
DMAC Fails-ARP Res = 0
IP Fails-ARP Req  = 0
IP Fails-ARP Res  = 0

```

If host 1 tries to send an ARP request with an IP address of 10.0.0.3, the packet is dropped and an error message is logged.

```

00:12:08: %SW_DAI-4-DHCP_SNOOPING_DENY: 2 Invalid ARPs (Req) on Ethernet2/3, vlan
1. ([0002.0002.0002/10.0.0.3/0000.0000.0000/0.0.0.0/02:42:35 UTC Fri Jul 13 2008])

```

The statistics display as follows:

```

switchA# show ip arp inspection statistics vlan 1
switchA#
Vlan : 1
-----
ARP Req Forwarded = 2
ARP Res Forwarded = 0
ARP Req Dropped   = 2
ARP Res Dropped   = 0
DHCP Drops        = 2
DHCP Permits      = 2
SMAC Fails-ARP Req = 0
SMAC Fails-ARP Res = 0
DMAC Fails-ARP Res = 0
IP Fails-ARP Req  = 0
IP Fails-ARP Res  = 0
switchA#

```

Configuring Device B

To enable DAI and configure Ethernet interface 1/4 on device B as trusted, follow these steps:

Procedure

Step 1 While logged into device B, verify the connection between device B and device A.

```

switchB# show cdp neighbors
Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  V - VoIP-Phone, D - Remotely-Managed-Device,
                  s - Supports-STP-Dispute

Device ID         Local Intrfce   Hldtme   Capability   Platform         Port ID
switchA           Ethernet1/4     120      R S I       WS-C2960-24TC   Ethernet2/3
switchB#

```

Step 2 Enable DAI on VLAN 1, and verify the configuration.


```

switchB# config t
switchB(config)# ip arp inspection vlan 1
switchB(config)# show ip arp inspection vlan 1
Source Mac Validation      : Disabled
Destination Mac Validation : Disabled
IP Address Validation      : Disabled
Vlan : 1
-----
Configuration      : Enabled
Operation State    : Active
switchB(config)#

```

Step 3 Configure Ethernet interface 1/4 as trusted.

```

switchB(config)# interface ethernet 1/4
switchB(config-if)# ip arp inspection trust
switchB(config-if)# exit
switchB(config)# exit
switchB# show ip arp inspection interface ethernet 1/4
  Interface      Trust State      Rate (pps)      Burst Interval
  -----      -
Ethernet1/4      Trusted           15              5
switchB#

```

Step 4 Verify the list of DHCP snooping bindings.

```

switchB# show ip dhcp snooping binding
MacAddress      IpAddress      LeaseSec      Type      VLAN      Interface
-----
00:01:00:01:00:01  10.0.0.2      4995         dhcp-snooping  1      Ethernet1/4
switchB#

```

Step 5 Check the statistics before and after DAI processes any packets.

```

switchB# show ip arp inspection statistics vlan 1
Vlan : 1
-----
ARP Req Forwarded = 0
ARP Res Forwarded = 0
ARP Req Dropped   = 0
ARP Res Dropped   = 0
DHCP Drops        = 0
DHCP Permits      = 0
SMAC Fails-ARP Req = 0
SMAC Fails-ARP Res = 0
DMAC Fails-ARP Res = 0
IP Fails-ARP Req  = 0
IP Fails-ARP Res  = 0
switchB#

```

If Host 2 sends out an ARP request with the IP address 10.0.0.2 and the MAC address 0001.0001.0001, the packet is forwarded and the statistics are updated.

```

switchB# show ip arp inspection statistics vlan 1
Vlan : 1
-----
ARP Req Forwarded = 1
ARP Res Forwarded = 0
ARP Req Dropped   = 0
ARP Res Dropped   = 0
DHCP Drops        = 0

```

```
DHCP Permits          = 1
SMAC Fails-ARP Req    = 0
SMAC Fails-ARP Res    = 0
DMAC Fails-ARP Res    = 0
IP Fails-ARP Req      = 0
IP Fails-ARP Res      = 0
switchB#
```

If Host 2 attempts to send an ARP request with the IP address 10.0.0.1, DAI drops the request and logs the following system message:

```
00:18:08: %SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Req) on Ethernet1/4, vlan
1. ([0001.0001.0001/10.0.0.1/0000.0000.0000/0.0.0.0/01:53:21 UTC Fri Jun 13 2008])
```

The statistics display as follows:

```
switchB# show ip arp inspection statistics vlan 1
Vlan : 1
-----
ARP Req Forwarded    = 1
ARP Res Forwarded    = 0
ARP Req Dropped      = 1
ARP Res Dropped      = 0
DHCP Drops           = 1
DHCP Permits         = 1
SMAC Fails-ARP Req   = 0
SMAC Fails-ARP Res   = 0
DMAC Fails-ARP Res   = 0
IP Fails-ARP Req     = 0
IP Fails-ARP Res     = 0
switchB#
```



CHAPTER 13

Configuring IP Source Guard

This chapter includes the following sections:

- [Information About IP Source Guard, on page 217](#)
- [Licensing Requirements for IP Source Guard, on page 218](#)
- [Prerequisites for IP Source Guard, on page 218](#)
- [Guidelines and Limitations for IP Source Guard, on page 218](#)
- [Default Settings for IP Source Guard, on page 219](#)
- [Configuring IP Source Guard, on page 219](#)
- [Displaying IP Source Guard Bindings, on page 221](#)
- [Configuration Example for IP Source Guard, on page 221](#)
- [Additional References for IP Source Guard, on page 221](#)

Information About IP Source Guard

IP Source Guard is a per-interface traffic filter that permits IP traffic only when the IP address and MAC address of each packet matches one of two sources of IP and MAC address bindings:

- Entries in the Dynamic Host Configuration Protocol (DHCP) snooping binding table.
- Static IP source entries that you configure.

Filtering on trusted IP and MAC address bindings helps prevent spoofing attacks, in which an attacker uses the IP address of a valid host to gain unauthorized network access. To circumvent IP Source Guard, an attacker would have to spoof both the IP address and the MAC address of a valid host.

You can enable IP Source Guard on Layer 2 interfaces that are not trusted by DHCP snooping. IP Source Guard supports interfaces that are configured to operate in access mode and trunk mode. When you initially enable IP Source Guard, all inbound IP traffic on the interface is blocked except for the following:

- DHCP packets, which DHCP snooping inspects and then forwards or drops, depending upon the results of inspecting the packet.
- IP traffic from static IP source entries that you have configured in the Cisco NX-OS device.

The device permits the IP traffic when DHCP snooping adds a binding table entry for the IP address and MAC address of an IP packet or when you have configured a static IP source entry.

The device drops IP packets when the IP address and MAC address of the packet do not have a binding table entry or a static IP source entry. For example, assume that the **show ip dhcp snooping binding** command displays the following binding table entry:

```

MacAddress          IpAddress    LeaseSec    Type          VLAN          Interface
-----
00:02:B3:3F:3B:99  10.5.5.2    6943       dhcp-snooping 10            Ethernet2/3

```

If the device receives an IP packet with an IP address of 10.5.5.2, IP Source Guard forwards the packet only if the MAC address of the packet is 00:02:B3:3F:3B:99.

Licensing Requirements for IP Source Guard

This table shows the licensing requirements for IP Source Guard.

Product	License Requirement
Cisco NX-OS	IP Source Guard requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For an explanation of the Cisco NX-OS licensing scheme, see the <i>Cisco NX-OS Licensing Guide</i> .

Prerequisites for IP Source Guard

IP Source Guard has the following prerequisite:

- You must enable the DHCP feature.

Guidelines and Limitations for IP Source Guard

IP Source Guard has the following configuration guidelines and limitations:

- IP Source Guard limits IP traffic on an interface to only those sources that have an IP-MAC address binding table entry or static IP source entry. When you first enable IP Source Guard on an interface, you may experience disruption in IP traffic until the hosts on the interface receive a new IP address from a DHCP server.
- IP Source Guard is dependent upon DHCP snooping to build and maintain the IP-MAC address binding table or upon manual maintenance of static IP source entries.

Default Settings for IP Source Guard

This table lists the default settings for IP Source Guard parameters.

Table 21: Default IP Source Guard Parameters

Parameters	Default
IP Source Guard	Disabled on each interface.
IP source entries	None. No static or default IP source entries exist by default.

Configuring IP Source Guard

Enabling or Disabling IP Source Guard on a Layer 2 Interface

You can enable or disable IP Source Guard on a Layer 2 interface. By default, IP Source Guard is disabled on all interfaces.

Before you begin

Ensure that the DHCP feature is enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet <i>slot/port</i> Example: switch(config)# interface ethernet 2/3 switch(config-if)#	Enters interface configuration mode for the specified interface.
Step 3	[no] ip verify source dhcp-snooping-vlan Example: switch(config-if)# ip verify source dhcp-snooping vlan	Enables IP Source Guard on the interface. The no option disables IP Source Guard on the interface.
Step 4	(Optional) show running-config dhcp Example: switch(config-if)# show running-config dhcp	Displays the running configuration for DHCP snooping, including the IP Source Guard configuration.

	Command or Action	Purpose
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Related Topics

[Adding or Removing a Static IP Source Entry](#), on page 220

Adding or Removing a Static IP Source Entry

You can add or remove a static IP source entry on a device. By default, there are no static IP source entries on a device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] ip source binding IP-address MAC-address vlan vlan-ID interface ethernet slot/port Example: <pre>switch(config)# ip source binding 10.5.22.17 001f.28bd.0013 vlan 100 interface ethernet 2/3</pre>	Creates a static IP source entry for the current interface, or if you use the no option, removes a static IP source entry.
Step 3	(Optional) show ip dhcp snooping binding [interface ethernet slot/port] Example: <pre>switch(config)# show ip dhcp snooping binding interface ethernet 2/3</pre>	Displays IP-MAC address bindings for the interface specified, including static IP source entries. Static entries appear with the term in the Type column.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Related Topics

[Enabling or Disabling IP Source Guard on a Layer 2 Interface](#), on page 219

[Displaying IP Source Guard Bindings](#), on page 221

Displaying IP Source Guard Bindings

Use the `show ip verify source` command to display IP-MAC address bindings.

Configuration Example for IP Source Guard

This example shows how to create a static IP source entry and then how to enable IP Source Guard on an interface.

```
ip source binding 10.5.22.17 001f.28bd.0013 vlan 100 interface ethernet 2/3
interface ethernet 2/3
  no shutdown
  ip verify source dhcp-snooping-vlan
```

Additional References for IP Source Guard

Related Documents

Related Topic	Document Title
IP Source Guard commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	<i>Cisco Nexus 5500 Series NX-OS Security Command Reference</i>

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—



CHAPTER 14

Configuring Control Plane Policing

This chapter contains the following sections:

- [Information About CoPP, on page 223](#)
- [Control Plane Protection, on page 224](#)
- [CoPP Policy Templates, on page 228](#)
- [CoPP and the Management Interface, on page 233](#)
- [Licensing Requirements for CoPP, on page 233](#)
- [Guidelines and Limitations for CoPP, on page 233](#)
- [Default Settings for CoPP, on page 234](#)
- [Configuring CoPP, on page 234](#)
- [Verifying the CoPP Configuration, on page 236](#)
- [Displaying the CoPP Configuration Status, on page 236](#)
- [Monitoring CoPP, on page 237](#)
- [Clearing the CoPP Statistics, on page 237](#)
- [Additional References for CoPP, on page 238](#)
- [Feature History for CoPP, on page 238](#)

Information About CoPP

Control Plane Policing (CoPP) protects the control plane and separates it from the data plane, which ensures network stability, reachability, and packet delivery.

This feature allows a policy map to be applied to the control plane. This policy map looks like a normal QoS policy and is applied to all traffic entering the switch from a non-management port. A common attack vector for network devices is the denial-of-service (DoS) attack, where excessive traffic is directed at the device interfaces.

The Cisco NX-OS device provides CoPP to prevent DoS attacks from impacting performance. Such attacks, which can be perpetrated either inadvertently or maliciously, typically involve high rates of traffic destined to the supervisor module or CPU itself.

The supervisor module divides the traffic that it manages into three functional components or planes:

Data plane

Handles all the data traffic. The basic functionality of a Cisco NX-OS device is to forward packets from one interface to another. The packets that are not meant for the switch itself are called the transit packets. These packets are handled by the data plane.

Control plane

Handles all routing protocol control traffic. These protocols, such as the Border Gateway Protocol (BGP) and the Open Shortest Path First (OSPF) Protocol, send control packets between devices. These packets are destined to router addresses and are called control plane packets.

Management plane

Runs the components meant for Cisco NX-OS device management purposes such as the command-line interface (CLI) and Simple Network Management Protocol (SNMP).

The supervisor module has both the management plane and control plane and is critical to the operation of the network. Any disruption or attacks to the supervisor module will result in serious network outages. For example, excessive traffic to the supervisor module could overload and slow down the performance of the entire Cisco NX-OS device. For example, a DoS attack on the supervisor module could generate IP traffic streams to the control plane at a very high rate, forcing the control plane to spend a large amount of time in handling these packets and preventing the control plane from processing genuine traffic.

Examples of DoS attacks include:

- Internet Control Message Protocol (ICMP) echo requests
- IP fragments
- TCP SYN flooding

These attacks can impact the device performance and have the following negative effects:

- Reduced service quality (such as poor voice, video, or critical applications traffic)
- High route processor or switch processor CPU utilization
- Route flaps due to loss of routing protocol updates or keepalives
- Unstable Layer 2 topology
- Slow or unresponsive interactive sessions with the CLI
- Processor resource exhaustion, such as the memory and buffers
- Indiscriminate drops of incoming packets

**Caution**

It is important to ensure that you protect the supervisor module from accidental or malicious attacks by configuring control plane protection.

Control Plane Protection

To protect the control plane, the Cisco NX-OS device segregates different packets destined for the control plane into different classes. Once these classes are identified, the Cisco NX-OS device polices the packets, which ensures that the supervisor module is not overwhelmed.

Control Plane Packet Types

Different types of packets can reach the control plane:

Receive packets

Packets that have the destination address of a router. The destination address can be a Layer 2 address (such as a router MAC address) or a Layer 3 address (such as the IP address of a router interface). These packets include router updates and keepalive messages. Multicast packets can also be in this category where packets are sent to multicast addresses that are used by a router.

Exception packets

Packets that need special handling by the supervisor module. For example, if a destination address is not present in the Forwarding Information Base (FIB) and results in a miss, the supervisor module sends an ICMP unreachable packet back to the sender. Another example is a packet with IP options set.

Redirected packets

Packets that are redirected to the supervisor module. Features like Dynamic Host Configuration Protocol (DHCP) snooping or dynamic Address Resolution Protocol (ARP) inspection redirect some packets to the supervisor module.

Glean packets

If a Layer 2 MAC address for a destination IP address is not present in the FIB, the supervisor module receives the packet and sends an ARP request to the host.

All of these different packets could be maliciously used to attack the control plane and overwhelm the Cisco NX-OS device. CoPP classifies these packets to different classes and provides a mechanism to individually control the rate at which the supervisor module receives these packets.

Classification for CoPP

For effective protection, the Cisco NX-OS device classifies the packets that reach the supervisor modules to allow you to apply different rate controlling policies based on the type of the packet. For example, you might want to be less strict with a protocol packet such as Hello messages but more strict with a packet that is sent to the supervisor module because the IP option is set.

Rate Controlling Mechanisms

Once the packets are classified, the Cisco NX-OS device has two different mechanisms to control the rate at which packets arrive at the supervisor module: policing and rate limiting.

Using hardware policers, you can define separate actions for traffic that conforms to or violates certain conditions. These actions can transmit the packet, mark down the packet, or drop the packet.

You can configure the following parameters for policing:

Committed information rate (CIR)

Desired bandwidth, specified as a bit rate.

Committed burst (BC)

Size of a traffic burst that can exceed the CIR within a given unit of time and not impact scheduling.

CoPP Class Maps

The following table shows the available class maps and their configurations.

Table 22: Class Map Configurations and Descriptions

Class Map	Configuration	Description
class-map type control-plane match-any copp-system-class-arp	match protocol arp match protocol nd	Class matches all ARP packets. Class matches all ARP packets and ND (NA, NS, RA, and RS) packets.
class-map type control-plane match-any copp-system-class-bgp	match protocol bgp	Class matches all BGP packets.
class-map type control-plane match-any copp-system-class-bridging	match protocol bridging	Class matches all STP and RSTP frames.
class-map type control-plane match-any copp-system-class-cdp	match protocol cdp	Class matches all CDP frames.
class-map type control-plane match-any copp-system-class-default	match protocol default	Class matches all frames. Used for the default policer.
class-map type control-plane match-any copp-system-class-dhcp	match protocol dhcp	Class matches all IPv4 DHCP packets Class matches all both IPv4 DHCP packets.
class-map type control-plane match-any copp-system-class-eigrp	match protocol eigrp match protocol eigrp6	Class matches all IPv4 EIGRP packets. Class matches both IPv4 and IPv6 EIGRP packets.
class-map type control-plane match-any copp-system-class-exception	match protocol exception	Class matches all IP packets that are treated as exception packets (except TTL exception, IP Fragment exception and Same Interface exception packets) for IP routing purposes, such as packets with a Martian destination address or with an MTU failure.
class-map type control-plane match-any copp-system-class-excp-ip-frag	match protocol ip_frag	Class matches all IP packets that are fragments. (These packets are treated as exception packets from an IP routing perspective).
class-map type control-plane match-any copp-system-class-excp-same-if	match protocol same-if	Class matches all IP packets that are treated as exception packets for IP routing. The packets are matched because they are received from the interface where their destination is supposed to be.

Class Map	Configuration	Description
class-map type control-plane match-any copp-system-class-excp-ttl	match protocol ttl	Class matches all packets that are treated as TTL exception packets (when TTL is 0) from a IP routing perspective.
class-map type control-plane match-any copp-system-class-fip	match protocol fip	Class matches all packets belonging to the FCoE Initialization Protocol.
class-map type control-plane match-any copp-system-class-glean	match protocol glean	Class matches all IP packets that cannot be routed to the next hop because the destination MAC information is unavailable.
class-map type control-plane match-any copp-system-class-hsrp-vrrp	match protocol hsrp_vrrp match protocol hsrp6	Class matches HSRP and VRRP packets. Class matches IPv4 HSRP, VRRP and IPv6 HSRP packets
class-map type control-plane match-any copp-system-class-icmp-echo	match protocol icmp_echo	Class matches all ICMP Echo (Ping) packets.
class-map type control-plane match-any copp-system-class-igmp	match protocol igmp	Class matches all IGMP packets.
class-map type control-plane match-any copp-system-class-isis	match protocol isis_dce	Class matches all ISIS protocol packets.
class-map type control-plane match-any copp-system-class-l3dest-miss	match protocol unicast	Class matches all unicast routed packets that did not find a destination in the FIB.
class-map type control-plane match-any copp-system-class-lacp	match protocol lacp	Class matches all Link Aggregation Control Protocol (LACP) frames.
class-map type control-plane match-any copp-system-class-lldp	match protocol lldp_dcx	Class matches all LLDP frames.
class-map type control-plane match-any copp-system-class-mcast-last-hop	match protocol mcast_last_hop	Class matches all IP multicast last hop packets.
class-map type control-plane match-any copp-system-class-mcast-miss	match protocol multicast	Class matches all IP multicast frames that could not be routed because they did not have an entry in the FIB.
class-map type control-plane match-any copp-system-class-mgmt	match protocol mgmt	Class matches all management-related frames, such as SNMP, HTTP, NTP, Telnet, and SSH.

Class Map	Configuration	Description
class-map type control-plane match-any copp-system-class-msdp	match protocol msdp	Class matches MSDP packets.
class-map type control-plane match-any copp-system-class-ospf	match protocol ospf match protocol ospfv3	Class matches OSPF and OSPFv3 Protocol packets.
class-map type control-plane match-any copp-system-class-pim-hello	match protocol pim	Class matches all PIM Hello packets.
class-map type control-plane match-any copp-system-class-pim-register	match protocol reg	Class matches all PIM Register packets.
class-map type control-plane match-any copp-system-class-rip	match protocol rip	Class matches all RIP packets.
class-map type control-plane match-any copp-system-class-rpf-fail	match protocol rpf_fail	Class matches all RPF failure packets.
class-map type control-plane match-any copp-system-class-udld	match protocol udld	Class matches all UDLD frames.

CoPP Policy Templates

When you bring up your Cisco NX-OS device for the first time, the Cisco NX-OS software installs the default `copp-system-policy` to protect the supervisor module from DoS attacks. You can choose the CoPP policy template for your deployment scenario by specifying CoPP policy options from the initial setup utility:

- Default CoPP Policy (`copp-system-policy-default`)
- Scaled Layer 2 CoPP Policy (`copp-system-policy-scaled-l2`)
- Scaled Layer 3 CoPP Policy (`copp-system-policy-scaled-l3`)
- Customized CoPP Policy (`copp-system-policy-customized`)

If you do not select an option or choose not to execute the setup utility, the Cisco NX-OS software applies the Default policing. Cisco recommends starting with the default policy and later modifying the CoPP policies as required.

The default `copp-system-policy-default` policy has optimized values suitable for basic device operations. You must add specific class and access-control list (ACL) rules that meet your DoS protection requirements.

You can change which CoPP policy is used by using the **service-policy input** *policy-name* command in the control plane configuration mode.

Default CoPP Policy

The `copp-system-policy-default` policy is applied to the switch by default. It has the classes with policer rates that should suit most network installations. You cannot modify this policy or the class maps associated with it. In addition, you cannot modify the class map configurations in this policy.

This policy has the following configuration:

```
policy-map type control-plane copp-system-policy-default
  class copp-system-class-igmp
    police cir 1024 kbps bc 65535 bytes
  class copp-system-class-pim-hello
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bridging
    police cir 20000 kbps bc 4800000 bytes
  class copp-system-class-arp
    police cir 1024 kbps bc 3600000 bytes
  class copp-system-class-dhcp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-mgmt
    police cir 12000 kbps bc 4800000 bytes
  class copp-system-class-lacp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-lldp
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-udld
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-isis
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-msdp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-cdp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-fip
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bgp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-eigrp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-exception
    police cir 64 kbps bc 4800000 bytes
  class copp-system-class-glean
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-hsrp-vrrp
    police cir 1024 kbps bc 256000 bytes
  class copp-system-class-icmp-echo
    police cir 64 kbps bc 3600000 bytes
  class copp-system-class-ospf
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-pim-register
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-rip
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-l3dest-miss
    police cir 64 kbps bc 256000 bytes
  class copp-system-class-mcast-miss
    police cir 256 kbps bc 3200000 bytes
  class copp-system-class-excp-ip-frag
    police cir 64 kbps bc 3200000 bytes
  class copp-system-class-excp-same-if
    police cir 64 kbps bc 3200000 bytes
  class copp-system-class-excp-ttl
```

```

    police cir 64 kbps bc 3200000 bytes
class copp-system-class-default
    police cir 2048 kbps bc 6400000 bytes

```

Scaled Layer 2 CoPP Policy

The `copp-system-policy-scaled` policy has most classes with policer rates that are same as the default policy. However, it has higher policer rates for IGMP and ISIS. You cannot modify this policy or the class maps associated with it. In addition, you cannot modify the class map configurations in this policy.

This policy has the following configuration:

```

policy-map type control-plane copp-system-policy-scaled-12
  class copp-system-class-igmp
    police cir 4096 kbps bc 264000 bytes
  class copp-system-class-pim-hello
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bridging
    police cir 20000 kbps bc 4800000 bytes
  class copp-system-class-arp
    police cir 1024 kbps bc 3600000 bytes
  class copp-system-class-dhcp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-mgmt
    police cir 12000 kbps bc 4800000 bytes
  class copp-system-class-lacp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-lldp
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-udld
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-isis
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-msdp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-cdp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-fip
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bgp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-eigrp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-exception
    police cir 64 kbps bc 4800000 bytes
  class copp-system-class-glean
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-hsrp-vrrp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-icmp-echo
    police cir 64 kbps bc 3600000 bytes
  class copp-system-class-ospf
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-pim-register
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-rip
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-l3dest-miss
    police cir 64 kbps bc 3200000 bytes
  class copp-system-class-mcast-miss
    police cir 256 kbps bc 3200000 bytes
  class copp-system-class-excp-ip-frag

```



```

    police cir 64 kbps bc 3200000 bytes
class copp-system-class-excp-same-if
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-excp-ttl
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-default
    police cir 2048 kbps bc 6400000 bytes

```

Scaled Layer 3 CoPP Policy

The `copp-system-policy-scaled-l3` policy has most classes with policer rates that are same as the default policy. However, it has higher policer rates for IGMP, ICMP Echo, ISIS, Mcast-miss, and Glean related classes. You cannot modify this policy or the class maps associated with it. In addition, you cannot modify the class map configurations in this policy.

This policy has the following configuration:

```

policy-map type control-plane copp-system-policy-scaled-l3
  class copp-system-class-igmp
    police cir 4096 kbps bc 264000 bytes
  class copp-system-class-pim-hello
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bridging
    police cir 20000 kbps bc 4800000 bytes
  class copp-system-class-arp
    police cir 4000 kbps bc 3600000 bytes
  class copp-system-class-dhcp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-mgmt
    police cir 12000 kbps bc 4800000 bytes
  class copp-system-class-lacp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-lldp
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-udld
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-isis
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-msdp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-cdp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-fip
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bgp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-eigrp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-exception
    police cir 64 kbps bc 4800000 bytes
  class copp-system-class-glean
    police cir 4000 kbps bc 4800000 bytes
  class copp-system-class-hsrp-vrrp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-icmp-echo
    police cir 4000 kbps bc 3600000 bytes
  class copp-system-class-ospf
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-pim-register
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-rip

```

```

    police cir 9600 kbps bc 4800000 bytes
class copp-system-class-l3dest-miss
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-mcast-miss
    police cir 4000 kbps bc 3200000 bytes
class copp-system-class-excp-ip-frag
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-excp-same-if
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-excp-ttl
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-default
    police cir 2048 kbps bc 6400000 bytes

```

Customizable CoPP Policy

The `copp-system-policy-customized` policy is configured identically to the default policy, but can be customized for different class map information rates and burst sizes.

You cannot add or delete any of the class maps configured in this policy.



Important

This policy is meant for advanced users. We recommend that you use extreme caution when configuring this policy and test it extensively before deploying it in your production network.

This policy has the following configuration:

```

policy-map type control-plane copp-system-policy-customized
  class copp-system-class-igmp
    police cir 1024 kbps bc 65535 bytes
  class copp-system-class-pim-hello
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bridging
    police cir 20000 kbps bc 4800000 bytes
  class copp-system-class-arp
    police cir 1024 kbps bc 3600000 bytes
  class copp-system-class-dhcp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-mgmt
    police cir 12000 kbps bc 4800000 bytes
  class copp-system-class-lacp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-lldp
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-udld
    police cir 2048 kbps bc 4800000 bytes
  class copp-system-class-isis
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-msdp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-cdp
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-fip
    police cir 1024 kbps bc 4800000 bytes
  class copp-system-class-bgp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-eigrp
    police cir 9600 kbps bc 4800000 bytes
  class copp-system-class-exception

```

```
    police cir 64 kbps bc 4800000 bytes
class copp-system-class-glean
    police cir 1024 kbps bc 4800000 bytes
class copp-system-class-hsrp-vrrp
    police cir 1024 kbps bc 4800000 bytes
class copp-system-class-icmp-echo
    police cir 64 kbps bc 3600000 bytes
class copp-system-class-ospf
    police cir 9600 kbps bc 4800000 bytes
class copp-system-class-pim-register
    police cir 9600 kbps bc 4800000 bytes
class copp-system-class-rip
    police cir 9600 kbps bc 4800000 bytes
class copp-system-class-l3dest-miss
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-mcast-miss
    police cir 256 kbps bc 3200000 bytes
class copp-system-class-excp-ip-frag
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-excp-same-if
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-excp-ttl
    police cir 64 kbps bc 3200000 bytes
class copp-system-class-default
    police cir 2048 kbps bc 6400000 bytes
```

CoPP and the Management Interface

The Cisco NX-OS device supports only hardware-based CoPP which does not support the management interface (mgmt0). The out-of-band mgmt0 interface connects directly to the CPU and does not pass through the in-band traffic hardware where CoPP is implemented.

On the mgmt0 interface, ACLs can be configured to give or deny access to a particular type of traffic.

Licensing Requirements for CoPP

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Guidelines and Limitations for CoPP

CoPP is a feature that is enabled by default in the switch. You cannot enable or disable CoPP.

- Only one control-plane policy can be applied at a time.
- Removing a CoPP policy applies the default CoPP policy. In this way, a CoPP policy is always applied.
- You cannot add or delete any classes or policies.
- You cannot change the order of the classes or remove a class from any policy.
- You cannot modify the default, the Scaled Layer-2, or the Scaled Layer 3 policies. However, you can modify the information rate and burst size of the classes in the customized policy.

- The customized policy configuration is the same as the default policy configuration, unless the customized policy has been modified.
- When upgrading from a previous release, the default CoPP policy is enabled by default on the switch.
- After modifying the customized policy or changing the applied policy, the statistical counters are reset.
- After you perform an ISSU, the statistical counters are reset.
- Cisco recommends that you use the default CoPP policy initially and then later determine which of the CoPP policies to use based on the data center and application requirements.
- Customizing CoPP is an ongoing process. CoPP must be configured according to the protocols and features used in your specific environment as well as the supervisor features that are required by the server environment. As these protocols and features change, CoPP must be modified.
- Cisco recommends that you continuously monitor CoPP. If drops occur, determine if CoPP dropped traffic unintentionally or in response to a malfunction or attack. In either event, analyze the situation and evaluate the need to use a different CoPP policy or modify the customized CoPP policy.
- All the traffic that you do not specify in the other class maps is put into the last class, the default class.
- The Cisco NX-OS software does not support egress CoPP or silent mode. CoPP is supported only on ingress (you cannot use the **service-policy output copp** command to the control plane interface).



Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Default Settings for CoPP

This table lists the default settings for CoPP parameters.

Table 23: Default CoPP Parameters Settings

Parameters	Default
Default policy	copp-system-policy-default
Scale factor value	1.00

Configuring CoPP

Applying a CoPP Policy to the Switch

You can apply one of the following CoPP policies to the switch:

- Default CoPP Policy (copp-system-policy-default).

- Scaled Layer 2 CoPP Policy (copp-system-policy-scaled-l2).
- Scaled Layer 3 CoPP Policy (copp-system-policy-scaled-l3).
- Customized CoPP Policy (copp-system-policy-customized).

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # control-plane	Enters control-plane mode.
Step 3	switch(config-cp) # service-policy input <i>policy-map-name</i>	Applies the specified CoPP policy map. The <i>policy-map-name</i> can be copp-system-policy-default, copp-system-policy-scaled-l2, copp-system-policy-scaled-l3, or copp-system-policy-customized.
Step 4	switch(config-cp) # copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to apply a CoPP policy to the device:

```
switch# configure terminal
switch(config)# control-plane
switch(config-cp) # service-policy input copp-system-policy-default
switch(config-cp) # copy running-config startup-config
```

Modifying the Customized CoPP Policy

You can only modify the information rates and burst sizes of the class maps configured in this policy.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# policy-map type control-plane copp-system-policy-customized	Enters configuration mode for the customized CoPP policy.
Step 3	switch(config-pmap)# class <i>class-map-name</i>	Specifies one of the 28 predefined class-maps listed in any CoPP predefined policy.
Step 4	switch(config-pmap-c)# police cir <i>rate-value</i> kbps bc <i>buffer-size</i> bytes	Configures the committed information rate (CIR) and committed burst size (BC). The range

	Command or Action	Purpose
		for cir is from 1 to 20480. The range for bc is from 1500 to 6400000.
Step 5	switch(config-pmap-c) # copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to modify the customized CoPP policy:

```
switch(config)# policy-map type control-plane copp-system-policy-customized
switch(config-pmap) # class copp-system-class-bridging
switch(config-pmap-c) # police cir 10000 kbps bc 2400000 bytes
```

Verifying the CoPP Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show policy-map type control-plane [expand] [name <i>policy-map-name</i>]	Displays the control plane policy map with associated class maps.
show policy-map interface control-plane	Displays the policy values with associated class maps and drops per policy or class map.
show class-map type control-plane [<i>class-map-name</i>]	Displays the control plane class map configuration, including the ACLs that are bound to this class map.

Displaying the CoPP Configuration Status

Procedure

	Command or Action	Purpose
Step 1	switch# show copp status	Displays the configuration status for the CoPP feature.

Example

This example shows how to display the CoPP configuration status:

```
switch# show copp status
```

Monitoring CoPP

Procedure

	Command or Action	Purpose
Step 1	switch# show policy-map interface control-plane	Displays packet-level statistics for all classes that are part of the applied CoPP policy. For example, Conformed and Violated packet counters. Statistics are specified in terms of OutPackets (packets admitted to the control plane) and DropPackets (packets dropped because of rate limiting).

Example

This example shows how to monitor CoPP:

```
switch# show policy-map interface control-plane
Control Plane

service-policy input: copp-system-policy-default

class-map copp-system-class-igmp (match-any)
match protocol igmp
police cir 1024 kbps , bc 65535 bytes
conformed 0 bytes; action: transmit
violated 0 bytes;
class-map copp-system-class-pim-hello (match-any)
match protocol pim
police cir 1024 kbps , bc 4800000 bytes
conformed 0 bytes; action: transmit
violated 0 bytes;
....
```

Clearing the CoPP Statistics

Procedure

	Command or Action	Purpose
Step 1	(Optional) switch# show policy-map interface control-plane	Displays the currently applied CoPP policy and per-class statistics.
Step 2	switch# clear copp statistics	Clears the CoPP statistics.

Example

This example shows how to clear the CoPP statistics for your installation:

```
switch# show policy-map interface control-plane
switch# clear copp statistics
```

Additional References for CoPP

This section provides additional information related to implementing CoPP.

Related Documents

Related Topic	Document Title
Licensing	<i>Cisco NX-OS Licensing Guide</i>
Command reference	<i>Cisco Nexus 5500 Series NX-OS Security Command Reference</i>

Feature History for CoPP

Table 24: Feature History for CoPP

Feature Name	Feature Information
CoPP	Introduced in 5.1(3)N1(1)
CoPP	Additional IPv6 support in 5.2(1)N1(1)



INDEX

- 802.1X [69](#), [72](#), [73](#), [74](#), [75](#), [76](#), [77](#), [78](#), [79](#), [80](#), [81](#), [84](#), [88](#), [89](#), [90](#), [91](#), [92](#), [94](#), [96](#)
 - authenticator PAEs [72](#)
 - configuration process [78](#)
 - configuring [78](#)
 - configuring AAA accounting methods [94](#)
 - configuring AAA authentication methods [79](#)
 - configuring on member ports [81](#)
 - controlling on interfaces [80](#)
 - default settings [77](#)
 - description [69](#)
 - disabling authentication [90](#)
 - disabling feature [91](#)
 - enabling feature [78](#)
 - enabling MAC authentication bypass [89](#)
 - enabling multiple hosts mode [88](#)
 - enabling periodic reauthentication on interfaces [84](#)
 - enabling single host mode [88](#)
 - example configuration [96](#)
 - guidelines [76](#)
 - licensing requirements [75](#)
 - limitations [76](#)
 - MAC authentication bypass [73](#)
 - monitoring [96](#)
 - multiple host support [74](#)
 - prerequisites [76](#)
 - setting interface maximum retransmission retry count [92](#)
 - single host support [74](#)
 - supported topologies [75](#)
 - verifying configuration [96](#)
 - 802.1X authentication [71](#), [72](#), [86](#), [93](#)
 - authorization states for ports [72](#)
 - changing timers on interfaces [86](#)
 - enabling RADIUS accounting [93](#)
 - initiation [71](#)
 - manually initializing [86](#)
 - 802.1X reauthentication [95](#)
 - setting maximum retry count on interfaces [95](#)
 - 802.1X supplicants [85](#)
 - manually reauthenticating [85](#)
- A**
- AAA [3](#), [7](#), [8](#), [9](#), [11](#), [12](#), [16](#), [20](#), [31](#), [79](#), [108](#)
 - accounting [7](#)
 - AAA (*continued*)
 - authentication [7](#)
 - benefits [8](#)
 - configuring authentication methods for 802.1X [79](#)
 - configuring console login [12](#)
 - configuring for Cisco TrustSec [108](#)
 - configuring for RADIUS servers [31](#)
 - configuring seed device for Cisco TrustSec [108](#)
 - default settings [20](#)
 - description [3](#)
 - enabling MSCHAP authentication [16](#)
 - example configuration [20](#)
 - prerequisites [11](#)
 - user login process [9](#)
 - verifying configurations [20](#)
 - AAA accounting [17](#), [94](#)
 - configuring default methods [17](#)
 - configuring methods for 802.1X [94](#)
 - AAA accounting logs [19](#)
 - clearing [19](#)
 - displaying [19](#)
 - AAA authorization [46](#)
 - configuring on TACACS+ servers [46](#)
 - AAA logins [14](#)
 - enabling authentication failure messages [14](#)
 - AAA protocols [7](#)
 - RADIUS [7](#)
 - TACACS+ [7](#)
 - AAA server groups [8](#)
 - description [8](#)
 - AAA servers [17](#), [19](#)
 - specifying SNMPv3 parameters [17](#), [19](#)
 - specifying user roles [19](#)
 - specifying user roles in VSAs [17](#)
 - AAA services [8](#)
 - configuration options [8](#)
 - remote [8](#)
 - accounting [7](#)
 - description [7](#)
 - ACL [136](#), [138](#)
 - processing order [136](#)
 - sequence numbers [138](#)
 - ACL implicit rules [137](#)
 - ACLs [135](#), [137](#), [140](#), [145](#), [154](#)
 - applications [135](#)

ACLs (*continued*)

- creating log entries for 145
 - guidelines 140
 - identifying traffic by protocols 137
 - licensing 140
 - limitations 140
 - prerequisites 140
 - types 135
 - VLAN 154
- authentication 7, 8, 9, 71
 - 802.1X 71
 - description 7
 - local 7
 - methods 8
 - remote 7
 - user login 9
 - authenticator PAs 72, 83
 - creating on an interface 83
 - description 72
 - removing from an interface 83
 - authorization 9, 49
 - user login 9
 - verifying commands 49

C

- Cisco 18, 23
 - vendor ID 18, 23
- Cisco TrustSec 99, 101, 104, 105, 106, 107, 108, 113, 114, 123, 129, 130
 - architecture 99
 - configuring 106
 - configuring AAA on seed device 108
 - configuring device credentials 107
 - configuring pause frame encryption and decryption on interfaces 113
 - default values 106
 - description 99
 - enabling 106
 - enabling (example) 130
 - environment data download 104
 - example configurations 130
 - guidelines 105
 - licensing 104
 - limitations 105
 - manually configuring SXP 123
 - prerequisites 105
 - SGACLs 101, 114
 - SGTs 101
 - verifying configuration 129
- Cisco TrustSec authentication 100, 108, 111, 130
 - configuring 108
 - configuring in manual mode 111
 - description 100
 - manual mode configuration examples 130
- Cisco TrustSec authorization 108
 - configuring 108
- Cisco TrustSec device credentials 101
 - description 101
- Cisco TrustSec device identities 100
 - description 100
- Cisco TrustSec environment data 104
 - download 104
- Cisco TrustSec policies 130
 - example enforcement configuration 130
- Cisco TrustSec seed devices 104, 108, 130
 - description 104, 108
 - example configuration 130
- Cisco TrustSec user credentials 101
 - description 101
- cisco-av-pair 17, 19
 - specifying AAA user parameters 17, 19
- class maps 225
 - CoPP 225
- clearing statistics 237
 - CoPP 237
- commands 49
 - disabling authorization verification 49
 - enabling authorization verification 49
- configuration status 236
 - CoPP 236
- control plane 234
 - policies 234
 - applying 234
- control plane class maps 236
 - verifying the configuration 236
- control plane policy maps 236
 - verifying the configuration 236
- control plane protection 224
 - CoPP 224
 - packet types 224
- control plane protection, classification 225
- control plane protection, CoPP 225
 - rate controlling mechanisms 225
- CoPP 223, 224, 225, 228, 233, 234, 236, 237, 238
 - class maps 225
 - clearing statistics 237
 - configuration status 236
 - control plane protection 224
 - control plane protection, classification 225
 - default settings 234
 - feature history 238
 - guidelines 233
 - information about 223
 - licensing 233
 - limitations 233
 - monitoring 237
 - policy templates 228
 - restrictions for management interfaces 233
 - verifying the configuration 236
- CoPP policies 229, 230, 231, 232, 234
 - applying 234
 - customized 232

CoPP policies (*continued*)

- default [229](#)
- scaled Layer 2 [230](#)
- scaled Layer 3 [231](#)
- CoPP policy [235](#)
 - customized [235](#)
 - modifying [235](#)
- CTS, *See* Cisco TrustSec
- customized CoPP policy [232, 235](#)
 - modifying [235](#)

D

- DAI [205, 206](#)
 - default settings [206](#)
 - guidelines [205](#)
 - limitations [205](#)
- default settings [178](#)
 - port security [178](#)
- default CoPP policy [229](#)
- default settings [20, 77, 206, 219, 234](#)
 - 802.1X [77](#)
 - AAA [20](#)
 - CoPP [234](#)
 - DAI [206](#)
 - IP Source Guard [219](#)
- device roles [69](#)
 - description for 802.1X [69](#)
- DHCP binding database, *See* DHCP snooping binding database
- DHCP Option 82 [183](#)
 - description [183](#)
- DHCP relay agent [186, 193, 194, 195, 196](#)
 - enabling or disabling [193](#)
 - enabling or disabling Option 82 [194](#)
 - enabling or disabling subnet broadcast support on a Layer 3 Interface [196](#)
 - enabling or disabling VRF support [195](#)
 - VRF support [186](#)
- DHCP relay binding database [187](#)
 - description [187](#)
- DHCP snooping [181, 183, 185, 187, 188](#)
 - binding database [183](#)
 - default settings [188](#)
 - description [181](#)
 - guidelines [187](#)
 - in a vPC environment [185](#)
 - limitations [187](#)
 - message exchange process [183](#)
 - Option 82 [183](#)
 - overview [181](#)
- DHCP snooping binding database [183](#)
 - described [183](#)
 - description [183](#)
 - entries [183](#)
 - See also* DHCP snooping binding database

- dynamic ARP inspection [201, 202, 203, 204](#)
 - ARP cache poisoning [201](#)
 - ARP requests [201](#)
 - ARP spoofing attack [201](#)
 - DHCP snooping binding database [202](#)
 - function of [202](#)
 - interface trust states [203](#)
 - logging of dropped packets [204](#)
 - network security issues and interface trust states [203](#)
- Dynamic Host Configuration Protocol snooping, *See* DHCP snooping

E

- examples [20](#)
 - AAA configurations [20](#)

F

- feature history [238](#)
 - CoPP [238](#)

G

- guidelines [140, 166, 187, 205, 233](#)
 - ACLs [140](#)
 - CoPP [233](#)
 - DAI [205](#)
 - DHCP snooping [187](#)
 - port security [166](#)

I

- IDs [18, 23](#)
 - Cisco vendor ID [18, 23](#)
- IP ACL implicit rules [137](#)
- IP ACL statistics [148](#)
 - clearing [148](#)
 - monitoring [148](#)
- IP ACLs [5, 135, 139, 143, 144, 146, 147](#)
 - applications [135](#)
 - applying as a Router ACL [146](#)
 - applying as port ACLs [147](#)
 - changing [143](#)
 - changing sequence numbers in [144](#)
 - description [5](#)
 - logical operation units [139](#)
 - logical operators [139](#)
 - removing [144](#)
 - types [135](#)
- IP Source Guard [219](#)
 - default settings [219](#)

L

licensing **75, 104, 140, 233**
 802.1X **75**
 ACLs **140**
 Cisco TrustSec **104**
 CoPP **233**

limitations **140, 166, 187, 205, 233**
 ACLs **140**
 CoPP **233**
 DAI **205**
 DHCP snooping **187**
 port security **166**

logging **145**
 creating ACL for **145**

logical operation units **139**
 IP ACLs **139**

logical operators **139**
 IP ACLs **139**

login **29**
 RADIUS servers **29**

LOU, *See* logical operation units

M

MAC ACL implicit rules **137**

MAC ACLs **149**
 ACLs **149**
 MAC **149**
 creating **149**

MAC addresses **161**
 learning **161**

MAC authentication **73, 89**
 bypass for 802.1X **73**
 enabling bypass in 802.1X **89**

management interfaces **233**
 CoPP restrictions **233**

monitoring **22, 32, 237**
 CoPP **237**
 RADIUS **22**
 RADIUS servers **32**

MSCHAP **16**
 enabling authentication **16**

N

new in this release **1**

P

policy templates **228**
 description **228**

port ACL **147**

port security **161, 164, 166, 178**
 default settings **178**
 guidelines **166**
 limitations **166**
 MAC address learning **161**
 MAC move **164**
 violations **164**

ports **72**
 authorization states for 802.1X **72**

preshared keys **38**
 TACACS+ **38**

privilege level support for TACACS+ authorization **49**
 configuring **49**

privilege roles **51**
 permitting or denying commands for **51**

R

RADIUS **4, 21, 22, 24, 30, 35, 36**
 configuring servers **24**
 configuring timeout intervals **30**
 configuring transmission retry counts **30**
 default settings **36**
 description **4**
 example configurations **36**
 monitoring **22**
 network environments **21**
 operations **22**
 prerequisites **24**
 statistics, displaying **35**

RADIUS accounting **93**
 enabling for 802.1X authentication **93**

RADIUS server groups **28**
 global source interfaces **28**

RADIUS server preshared keys **26**

RADIUS servers **29, 30, 31, 34, 35, 36**
 allowing users to specify at login **29**
 configuring AAA for **31**
 configuring timeout interval **30**
 configuring transmission retry count **30**
 deleting hosts **34**
 displaying statistics **35**
 example configurations **36**
 manually monitoring **34**

RADIUS statistics **35**
 clearing **35**

RADIUS, global preshared keys **25**

RADIUS, periodic server monitoring **32**

RADIUS, server hosts **24**
 configuring **24**

rate controlling mechanisms **225**
 control plane protection, CoPP **225**

RBACL **121**
 clearing statistics **121**
 displaying statistics **121**

- RBACL (*continued*)
 - enabling statistics [121](#)
- RBACL logging [119](#)
 - enabling [119](#)
- remote devices [63](#)
 - connecting to using SSH [63](#)
- router ACLs [146](#)
- rules [137](#)
 - implicit [137](#)

S

- scaled Layer 2 CoPP policy [230](#)
- scaled Layer 3 CoPP policy [231](#)
- secure MAC addresses [161](#)
 - learning [161](#)
- security [161, 234](#)
 - policies [234](#)
 - applying [234](#)
 - port [161](#)
 - MAC address learning [161](#)
- security group access lists, *See* SGACLs
- security group tag, *See* SGT
- server groups [8](#)
- servers [29](#)
 - RADIUS [29](#)
- SGACL policies [119, 121, 122](#)
 - clearing [122](#)
 - displaying downloaded policies [121](#)
 - manually configuring [119](#)
- SGACL policy enforcement [115](#)
 - enabling on VLANs [115](#)
- SGACLs [101, 114, 131](#)
 - configuring [114](#)
 - description [101](#)
 - example manual configuration [131](#)
 - example SGT mapping configuration [131](#)
- SGACLs policies [121](#)
 - refreshing downloaded policies [121](#)
- SGT Exchange Protocol, *See* SXP
- SGTs [101, 103, 116, 117, 118, 131](#)
 - description [101](#)
 - example mapping configuration [131](#)
 - manually configuring [116](#)
 - manually configuring address-to-SGACL mapping [117, 118](#)
 - propagation with SXP [103](#)
- SNMPv3 [17, 19](#)
 - specifying AAA parameters [17](#)
 - specifying parameters for AAA servers [19](#)
- source interfaces [28, 44](#)
 - RADIUS server groups [28](#)
 - TACACS+ server groups [44](#)
- SSH [4](#)
 - description [4](#)
- SSH clients [59](#)
- SSH server keys [59](#)

- SSH servers [59](#)
- SSH sessions [63, 65](#)
 - clearing [65](#)
 - connecting to remote devices [63](#)
- statistics [56, 121, 148](#)
 - clearing [148](#)
 - for RBACL [121](#)
 - monitoring [148](#)
 - TACACS+ [56](#)
- SXP [103, 123, 124, 126, 127, 128](#)
 - changing retry periods [128](#)
 - configuration process [123](#)
 - configuring default passwords [126](#)
 - configuring default source IP addresses [127](#)
 - configuring manually [123](#)
 - configuring peer connections [124](#)
 - enabling [124](#)
 - SGT propagation [103](#)
- SXP connections [132](#)
 - example manual configuration [132](#)

T

- TACACS+ [4, 37, 38, 39, 40, 49, 52, 56, 57](#)
 - advantages over RADIUS [37](#)
 - configuring [40](#)
 - configuring global timeout interval [52](#)
 - description [4, 37](#)
 - displaying statistics [56](#)
 - example configurations [57](#)
 - field descriptions [57](#)
 - global preshared keys [38](#)
 - limitations [40](#)
 - prerequisites [39](#)
 - preshared key [38](#)
 - user login operation [38](#)
 - verifying command authorization [49](#)
 - verifying configuration [56](#)
- TACACS+ command authorization [47, 48](#)
 - configuring [47](#)
 - testing [48](#)
- TACACS+ server groups [44](#)
 - global source interfaces [44](#)
- TACACS+ servers [40, 52, 53, 55, 56, 57](#)
 - configuring hosts [40](#)
 - configuring TCP ports [53](#)
 - configuring timeout interval [52](#)
 - displaying statistics [56](#)
 - field descriptions [57](#)
 - manually monitoring [55](#)
 - verifying configuration [56](#)
- TCP ports [53](#)
 - TACACS+ servers [53](#)
- Telnet [4](#)
 - description [4](#)

Telnet server [66](#)
 enabling [66](#)
 reenabling [66](#)
Telnet servers [60](#)
Telnet sessions [66, 67](#)
 clearing [67](#)
 connecting to remote devices [66](#)

U

user login [9](#)
 authentication process [9](#)
 authorization process [9](#)

user roles [17, 19](#)
 specifying on AAA servers [17, 19](#)

V

vendor-specific attributes [18](#)
VLAN ACLs [154](#)
 information about [154](#)
vPCs [185](#)
 and DHCP snooping [185](#)
VSAs [18](#)
 format [18](#)
 protocol options [18](#)
 support description [18](#)