NetFlow Lite Configuration Guide, Cisco IOS Release 15.2(2)E (Catalyst 2960-X Switch)

First Published: June 05, 2014

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-32550-01

CONTENTS

Preface

This book describes configuration information and examples for NetFlow Lite on the switch.

Document Conventions

This document uses the following conventions:

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

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

Reader Alert Conventions

This document may use the following conventions for reader alerts:

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

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

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

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

Warning IMPORTANT SAFETY INSTRUCTIONS

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

SAVE THESE INSTRUCTIONS

Related Documentation

Note Before installing or upgrading the switch, refer to the switch release notes.

- Catalyst 2960-X Switch documentation, located at:

  http://www.cisco.com/go/cat2960x_docs

- Cisco SFP and SFP+ modules documentation, including compatibility matrixes, located at:

  http://www.cisco.com/en/US/products/hw/modules/ps5455/tsd_products_support_series_home.html

- Error Message Decoder, located at:

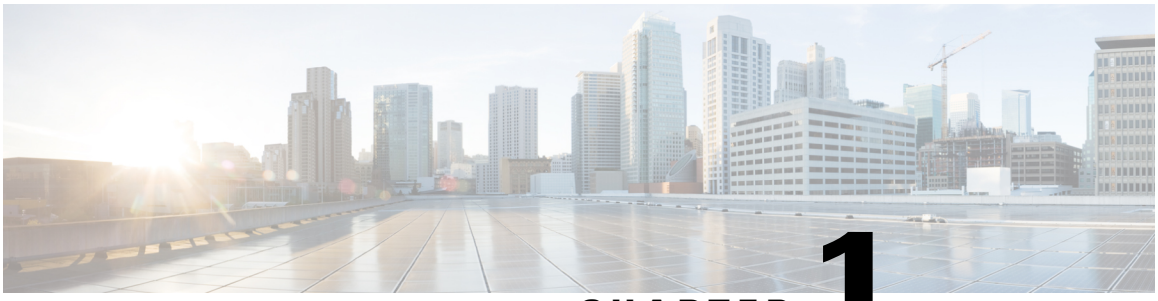
  https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Obtaining Documentation and Submitting a Service Request

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

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

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



CHAPTER 1

Using the Command-Line Interface

Information About Using the Command-Line Interface

Command Modes

The Cisco IOS user interface is divided into many different modes. The commands available to you depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands available for each command mode.

You can start a CLI session through a console connection, through Telnet, an SSH, or by using the browser.

When you start a session, you begin in user mode, often called user EXEC mode. Only a limited subset of the commands are available in user EXEC mode. For example, most of the user EXEC commands are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The user EXEC commands are not saved when the switch reboots.

To have access to all commands, you must enter privileged EXEC mode. Normally, you must enter a password to enter privileged EXEC mode. From this mode, you can enter any privileged EXEC command or enter global configuration mode.

Using the configuration modes (global, interface, and line), you can make changes to the running configuration. If you save the configuration, these commands are stored and used when the switch reboots. To access the various configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and line configuration mode .

This table describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode.

Table 1: Command Mode Summary

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session using Telnet, SSH, or console.	`Switch>`	Enter **logout** or **quit**.	Use this mode to • Change terminal settings. • Perform basic tests. • Display system information.
Privileged EXEC	While in user EXEC mode, enter the **enable** command.	`Switch#`	Enter **disable** to exit.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.
Global configuration	While in privileged EXEC mode, enter the **configure** command.	`Switch(config)#`	To exit to privileged EXEC mode, enter **exit** or **end**, or press **Ctrl-Z**.	Use this mode to configure parameters that apply to the entire switch.
VLAN configuration	While in global configuration mode, enter the **vlan** *vlan-id* command.	`Switch(config-vlan)#`	To exit to global configuration mode, enter the **exit** command. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure VLAN parameters. When VTP mode is transparent, you can create extended-range VLANs (VLAN IDs greater than 1005) and save configurations in the switch startup configuration file.
Interface configuration	While in global configuration mode, enter the **interface** command (with a specific interface).	`Switch(config-if)#`		Use this mode to configure parameters for the Ethernet ports.

Mode	Access Method	Prompt	Exit Method	About This Mode
			To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	
Line configuration	While in global configuration mode, specify a line with the **line vty** or **line console** command.	`Switch(config-line)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the terminal line.

Understanding Abbreviated Commands

You need to enter only enough characters for the switch to recognize the command as unique.

This example shows how to enter the **show configuration** privileged EXEC command in an abbreviated form:

```
Switch# show conf
```

No and Default Forms of Commands

Almost every configuration command also has a **no** form. In general, use the **no** form to disable a feature or function or reverse the action of a command. For example, the **no shutdown** interface configuration command reverses the shutdown of an interface. Use the command without the keyword **no** to reenable a disabled feature or to enable a feature that is disabled by default.

Configuration commands can also have a **default** form. The **default** form of a command returns the command setting to its default. Most commands are disabled by default, so the **default** form is the same as the **no** form. However, some commands are enabled by default and have variables set to certain default values. In these cases, the **default** command enables the command and sets variables to their default values.

CLI Error Messages

This table lists some error messages that you might encounter while using the CLI to configure your switch.

Table 2: Common CLI Error Messages

Error Message	Meaning	How to Get Help
`% Ambiguous command: "show con"`	You did not enter enough characters for your switch to recognize the command.	Reenter the command followed by a question mark (?) without any space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Incomplete command.`	You did not enter all of the keywords or values required by this command.	Reenter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Invalid input detected at '^' marker.`	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all of the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Configuration Logging

You can log and view changes to the switch configuration. You can use the Configuration Change Logging and Notification feature to track changes on a per-session and per-user basis. The logger tracks each configuration command that is applied, the user who entered the command, the time that the command was entered, and the parser return code for the command. This feature includes a mechanism for asynchronous notification to registered applications whenever the configuration changes. You can choose to have the notifications sent to the syslog.

Note Only CLI or HTTP changes are logged.

Using the Help System

You can enter a question mark (?) at the system prompt to display a list of commands available for each command mode. You can also obtain a list of associated keywords and arguments for any command.

SUMMARY STEPS

1. **help**
2. *abbreviated-command-entry* **?**
3. *abbreviated-command-entry* <Tab>
4. **?**
5. *command* **?**
6. *command keyword* **?**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**help** **Example:** `Switch# help`	Obtains a brief description of the help system in any command mode.

Step 2	*abbreviated-command-entry* **?**
	
	Example:
	`Switch# di?`
	`dir disable disconnect`
		Obtains a list of commands that begin with a particular character string.

Step 3	*abbreviated-command-entry* <Tab>
	
	Example:
	`Switch# sh conf<tab>`
	`Switch# show configuration`
		Completes a partial command name.

Step 4	**?** **Example:** `Switch> ?`	Lists all commands available for a particular command mode.
Step 5	*command* **?** **Example:** `Switch> show ?`	Lists the associated keywords for a command.

Step 6	*command keyword* **?**
	
	Example:
	`Switch(config)# cdp holdtime ?`
	`<10-255> Length of time (in sec) that receiver must keep this packet`
		Lists the associated arguments for a keyword.

How to Use the CLI to Configure Features

Configuring the Command History

The software provides a history or record of commands that you have entered. The command history feature is particularly useful for recalling long or complex commands or entries, including access lists. You can customize this feature to suit your needs.

Changing the Command History Buffer Size

By default, the switch records ten command lines in its history buffer. You can alter this number for a current terminal session or for all sessions on a particular line. This procedure is optional.

SUMMARY STEPS

1. **terminal history** [**size** *number-of-lines*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal history** [**size** *number-of-lines*] **Example:** `Switch# terminal history size 200`	Changes the number of command lines that the switch records during the current terminal session in privileged EXEC mode. You can configure the size from 0 to 256.

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in this table. These actions are optional.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

SUMMARY STEPS

1. **Ctrl-P** or use the **up arrow** key
2. **Ctrl-N** or use the **down arrow** key
3. **show history**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Ctrl-P** or use the **up arrow** key	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Step 2	**Ctrl-N** or use the **down arrow** key	Returns to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the up arrow key. Repeat the key sequence to recall successively more recent commands.
Step 3	**show history** **Example:** `Switch# show history`	Lists the last several commands that you just entered in privileged EXEC mode. The number of commands that appear is controlled by the setting of the **terminal history** global configuration command and the **history** line configuration command.

Disabling the Command History Feature

The command history feature is automatically enabled. You can disable it for the current terminal session or for the command line. This procedure is optional.

SUMMARY STEPS

1. **terminal no history**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal no history** **Example:** `Switch# terminal no history`	Disables the feature during the current terminal session in privileged EXEC mode.

Enabling and Disabling Editing Features

Although enhanced editing mode is automatically enabled, you can disable it and reenable it.

SUMMARY STEPS

1. **terminal editing**
2. **terminal no editing**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal editing** **Example:** `Switch# terminal editing`	Reenables the enhanced editing mode for the current terminal session in privileged EXEC mode.
Step 2	**terminal no editing** **Example:** `Switch# terminal no editing`	Disables the enhanced editing mode for the current terminal session in privileged EXEC mode.

Editing Commands Through Keystrokes

The keystrokes help you to edit the command lines. These keystrokes are optional.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

Table 3: Editing Commands

Editing Commands	Description
Ctrl-B or use the **left arrow** key	Moves the cursor back one character.
Ctrl-F or use the **right arrow** key	Moves the cursor forward one character.
Ctrl-A	Moves the cursor to the beginning of the command line.
Ctrl-E	Moves the cursor to the end of the command line.
Esc B	Moves the cursor back one word.
Esc F	Moves the cursor forward one word.
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.
Delete or **Backspace** key	Erases the character to the left of the cursor.
Ctrl-D	Deletes the character at the cursor.

Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-U or **Ctrl-X**	Deletes all characters from the cursor to the beginning of the command line.
Ctrl-W	Deletes the word to the left of the cursor.
Esc D	Deletes from the cursor to the end of the word.
Esc C	Capitalizes at the cursor.
Esc L	Changes the word at the cursor to lowercase.
Esc U	Capitalizes letters from the cursor to the end of the word.
Ctrl-V or **Esc Q**	Designates a particular keystroke as an executable command, perhaps as a shortcut.
Return key	Scrolls down a line or screen on displays that are longer than the terminal screen can display. **Note** The More prompt is used for any output that has more lines than can be displayed on the terminal screen, including **show** command output. You can use the **Return** and **Space** bar keystrokes whenever you see the More prompt.
Space bar	Scrolls down one screen.
Ctrl-L or **Ctrl-R**	Redisplays the current command line if the switch suddenly sends a message to your screen.

Editing Command Lines That Wrap

You can use a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command. The keystroke actions are optional.

To scroll back to the beginning of the command entry, press **Ctrl-B** or the left arrow key repeatedly. You can also press **Ctrl-A** to immediately move to the beginning of the line.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

The following example shows how to wrap a command line that extends beyond a single line on the screen.

SUMMARY STEPS

1. **access-list**
2. **Ctrl-A**
3. **Return** key

DETAILED STEPS

	Command or Action	Purpose

Step 1	**access-list**
	
	Example:
	`Switch(config)# access-list 101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35`
	`Switch(config)# $ 101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35 255.25`
	`Switch(config)# $t tcp 10.15.22.25 255.255.255.0 131.108.1.20 255.255.255.0 eq`
	`Switch(config)# $15.22.25 255.255.255.0 10.15.22.35 255.255.255.0 eq 45`
		Displays the global configuration command entry that extends beyond one line.
		
		When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign ($) shows that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.

Step 2	**Ctrl-A** **Example:** `Switch(config)# access-list 101 permit tcp 10.15.22.25 255.255.255.0 10.15.2$`	Checks the complete syntax. The dollar sign ($) appears at the end of the line to show that the line has been scrolled to the right.
Step 3	**Return** key	Execute the commands. The software assumes that you have a terminal screen that is 80 columns wide. If you have a different width, use the **terminal width** privileged EXEC command to set the width of your terminal. Use line wrapping with the command history feature to recall and modify previous complex command entries.

Searching and Filtering Output of show and more Commands

You can search and filter the output for **show** and **more** commands. This is useful when you need to sort through large amounts of output or if you want to exclude output that you do not need to see. Using these commands is optional.

SUMMARY STEPS

1. {**show** | **more**} *command* | {**begin** | **include** | **exclude**} *regular-expression*

DETAILED STEPS

	Command or Action	Purpose

Step 1	{**show** | **more**} *command* | {**begin** | **include** | **exclude**} *regular-expression*
	
	Example:
	`Switch# show interfaces | include protocol`
	`Vlan1 is up, line protocol is up`
	`Vlan10 is up, line protocol is down`
	`GigabitEthernet1/0/1 is up, line protocol is down`
	`GigabitEthernet1/0/2 is up, line protocol is up`
		Searches and filters the output.
		
		Expressions are case sensitive. For example, if you enter **| exclude output**, the lines that contain **output** are not displayed, but the lines that contain **output** appear.

Accessing the CLI on a Switch Stack

You can access the CLI through a console connection, through Telnet, a SSH, or by using the browser.

You manage the switch stack and the stack member interfaces through the active switchstack master. You cannot manage stack members on an individual switch basis. You can connect to the active switchstack master through the console port or the Ethernet management port of one or more stack members. Be careful with using multiple CLI sessions on the active switchstack master. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible to lose track of the session from which you entered commands.

Note We recommend using one CLI session when managing the switch stack.

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

To debug a specific stack member, you can start a CLI session from the stack master by using the **session** *stack-member-number* privileged EXEC command. The stack member number is appended to the system prompt. For example, *Switch-2#* is the prompt for stack member 2 where the system prompt for the stack master is Switch. Only the **show** and **debug** commands are available in a CLI session to a specific stack member. You can also use the **remote command** *stack-member-number LINE* privileged EXEC command on the stack master to enable debugging on a member switch without first starting a session.

Accessing the CLI Through a Console Connection or Through Telnet

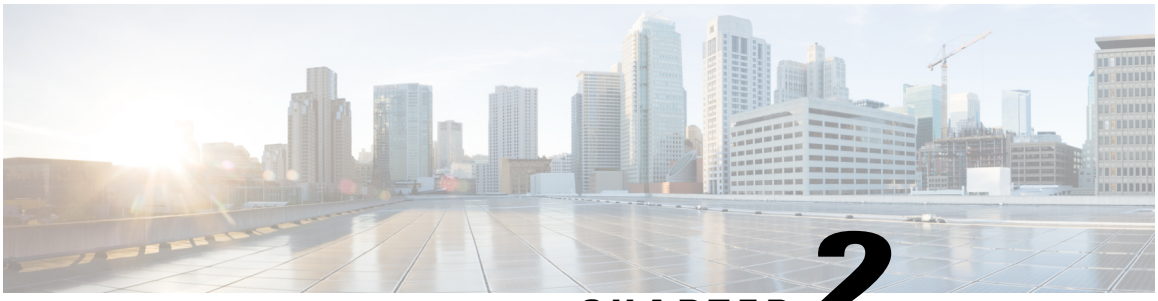
Before you can access the CLI, you must connect a terminal or a PC to the switch console or connect a PC to the Ethernet management port and then power on the switch, as described in the hardware installation guide that shipped with your switch.

If your switch is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your switch must first be configured for this type of access.

You can use one of these methods to establish a connection with the switch:

- Connect the switch console port to a management station or dial-up modem, or connect the Ethernet management port to a PC. For information about connecting to the console or Ethernet management port, see the switch hardware installation guide.
- Use any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The switch must have network connectivity with the Telnet or SSH client, and the switch must have an enable secret password configured.
    - The switch supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.
    - The switch supports up to five simultaneous secure SSH sessions.

  After you connect through the console port, through the Ethernet management port, through a Telnet session or through an SSH session, the user EXEC prompt appears on the management station.



CHAPTER 2

Configuring NetFlow Lite

Finding Feature Information

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

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for NetFlow Lite

NetFlow Lite is only supported on a Catalyst 2960-X Switch with a LAN Base license and on a Catalyst 2960-XR Switch with an IP Lite license. Catalyst 2960-XR is not stackable with the Catalyst 2960-X platform.

The following two targets for attaching a NetFlow Lite monitor are supported:

- Port—Monitor attachment is only supported on physical interfaces and not on logical interfaces, such as EtherChannels. The physical interface could be a routed port or a switched port.
- VLAN—Monitor attachment is supported on VLAN interfaces only (SVI) and not on a Layer 2 VLAN.

Restrictions for NetFlow Lite

The following are restrictions for NetFlow Lite:

- Monitor restrictions:
  - Monitor attachment is only supported in the ingress direction.
  - One monitor per interface is supported, although multiple exporters per interface are supported.
  - Only permanent and normal cache is supported for the monitor; immediate cache is not supported.
  - Changing any monitor parameter will not be supported when it is applied on any of the interfaces or VLANs.
  - When both the port and VLANs have monitors attached, then VLAN monitor will overwrite the port monitor for traffic coming on the port.
  - Flow monitor type and traffic type (type means IPv4, IPv6, and data link) should be same for the flows to be created.
  - You cannot attach an IP and port-based monitor to an interface at the same time on the switch. A 48-port switch supports a maximum of 48 monitors (IP or port-based) and for 256 SVIs, you can configure up to 256 monitors (IP or port-based).
  - When running the **show flow monitor** *flow_name* **cache** command, the switch displays cache information from an earlier switch software version (Catalyst 2960-S) with all fields entered as zero. Ignore these fields, as they are inapplicable to the switch.
- Sampler restrictions:
  - Only sampled NetFlow is supported.
  - For both port and VLANS, a total of only 4 samplers (random or deterministic) are supported on the switch.
  - The sampling minimum rate for both modes is 1 out of 32 flows, and the sampling maximum rate for both modes is 1 out of 1022 flows.
  - You must associate a sampler with a monitor while attaching it to an interface. Otherwise, the command will be rejected. Use the **ip flow monitor** *monitor_name* sampler s*ampler_name* **input** interface configuration command to perform this task.
  - When you attach a monitor using a deterministic sampler, every attachment with the same sampler uses one new free sampler from the switch (hardware) out of 4 available samplers. You are not allowed to attach a monitor with any sampler, beyond 4 attachments.

    When you attach a monitor using a random sampler, only the first attachment uses a new sampler from the switch (hardware). The remainder of all of the attachments using the same sampler, share the same sampler.

    Because of this behavior, when using a deterministic sampler, you can always make sure that the correct number of flows are sampled by comparing the sampling rate and what the switch sends. If the same random sampler is used with multiple interfaces, flows from any interface can always be sampled, and flows from other interfaces can always be skipped.
- Stacking Restrictions:

- The switch supports homogeneous stacking and mixed stacking. Mixed stacking is supported only with the Catalyst 2960-S switches. A homogenous stack can have up to eight stack members, while a mixed stack can have up to four stack members. All switches in a switch stack must be running the LAN Base image.
- The switch supports NetFlow Lite running on a mixed stack configuration, where both Catalyst 2960-X and Catalyst 2960-S switches reside in the same stack. But in such a mixed stack configuration, the master switch must always be a Catalyst 2960-X switch. The Catalyst 2960-S switch must never be the master switch in this type of mixed stack configuration.
- Each switch in a stack (hardware) can support the creation of a maximum of 16,000 flows at any time. But as the flows are periodically pushed to the software cache, the software cache can hold a much larger amount of flows (1048 Kb flows). From the hardware flow cache, every 20 seconds (termed as poll timer), 200 flows (termed as poll entries) are pushed to software.
  - Use the **remote command all show platform hulc-fnf poll** command to report on each switch's current NetFlow polling parameters.
  - Use the **show platform hulc-fnf poll** command to report on the master switch's current NetFlow polling parameters.

- Network flows and statistics are collected at the line rate.
- ACL-based NetFlow is not supported.
- Only NetFlow Version 9 is supported for Flexible NetFlow exporter using the *export-protocol* command option. If you configure NetFlow Version 5, this version will be accepted, but the NetFlow Version 5 export functionality is neither currently available nor supported.
- The switch supports homogeneous stacking, but does not support mixed stacking.

Information About NetFlow Lite

NetFlow Lite Overview

NetFlow Lite uses flows to provide statistics for accounting, network monitoring, and network planning.

A flow is a unidirectional stream of packets that arrives on a source interface and has the same values for the keys. A key is an identified value for a field within the packet. You create a flow using a flow record to define the unique keys for your flow.

The switch supports the NetFlow Lite feature that enables enhanced network anomalies and security detection. NetFlow Lite allows you to define an optimal flow record for a particular application by selecting the keys from a large collection of predefined fields.

All key values must match for the packet to count in a given flow. A flow might gather other fields of interest, depending on the export record version that you configure. Flows are stored in the NetFlow Lite cache.

You can export the data that NetFlow Lite gathers for your flow by using an exporter and export this data to a remote system such as a NetFlow Lite collector. The NetFlow Lite collector can use an IPv4 address.

You define the size of the data that you want to collect for a flow using a monitor. The monitor combines the flow record and exporter with the NetFlow Lite cache information.

Flexible NetFlow Components

Flexible NetFlow consists of components that can be used together in several variations to perform traffic analysis and data export. The user-defined flow records and the component structure of Flexible NetFlow facilitates the creation of various configurations for traffic analysis and data export on a networking device with a minimum number of configuration commands. Each flow monitor can have a unique combination of flow record, flow exporter, and cache type. If you change a parameter such as the destination IP address for a flow exporter, it is automatically changed for all the flow monitors that use the flow exporter. The same flow monitor can be used in conjunction with different flow samplers to sample the same type of network traffic at different rates on different interfaces. The following sections provide more information on Flexible NetFlow components:

Flow Records

In Flexible NetFlow a combination of key and nonkey fields is called a record. Flexible NetFlow records are assigned to Flexible NetFlow flow monitors to define the cache that is used for storing flow data.

A flow record defines the keys that Flexible NetFlow uses to identify packets in the flow, as well as other fields of interest that Flexible NetFlow gathers for the flow. You can define a flow record with any combination of keys and fields of interest. The switch supports a rich set of keys. A flow record also defines the types of counters gathered per flow. You can configure 64-bit packet or byte counters. The switch enables the following match fields as the defaults when you create a flow record:

- match datalink—Layer 2 attributes
- match ipv4—IPv4 attributes
- match ipv6—IPv6 attributes
- match transport—Transport layer fields
- match wireless—Wireless fields

Related Topics

Creating a Flow Record, on page 24

Example: Configuring a Flow, on page 38

NetFlow Predefined Records

Flexible NetFlow includes several predefined records that you can use to start monitoring traffic in your network. The predefined records are available to help you quickly deploy Flexible NetFlow and are easier to use than user-defined flow records. You can choose from a list of already defined records that may meet the needs for network monitoring. As Flexible NetFlow evolves, popular user-defined flow records will be made available as predefined records to make them easier to implement.

The predefined records ensure backward compatibility with your existing NetFlow collector configurations for the data that is exported. Each of the predefined records has a unique combination of key and nonkey fields that offer you the built-in ability to monitor various types of traffic in your network without customizing Flexible NetFlow on your router.

Two of the predefined records (NetFlow original and NetFlow IPv4/IPv6 original output), which are functionally equivalent, emulate original (ingress) NetFlow and the Egress NetFlow Accounting feature in original NetFlow,

respectively. Some of the other Flexible NetFlow predefined records are based on the aggregation cache schemes available in original NetFlow. The Flexible NetFlow predefined records that are based on the aggregation cache schemes available in original NetFlow do not perform aggregation. Instead each flow is tracked separately by the predefined records.

User-Defined Records

Flexible NetFlow enables you to define your own records for a Flexible NetFlow flow monitor cache by specifying the key and nonkey fields to customize the data collection to your specific requirements. When you define your own records for a Flexible NetFlow flow monitor cache, they are referred to as *user-defined records*. The values in nonkey fields are added to flows to provide additional information about the traffic in the flows. A change in the value of a nonkey field does not create a new flow. In most cases the values for nonkey fields are taken from only the first packet in the flow. Flexible NetFlow enables you to capture counter values such as the number of bytes and packets in a flow as nonkey fields.

Flexible NetFlow adds a new Version 9 export format field type for the header and packet section types. Flexible NetFlow will communicate to the NetFlow collector the configured section sizes in the corresponding Version 9 export template fields. The payload sections will have a corresponding length field that can be used to collect the actual size of the collected section.

NetFlow Lite Match Parameters

You can match these key fields for the flow record:

- IPv4 or IPv6 destination address
- Datalink fields (source and destination MAC address, and MAC ethertype (type of networking protocol)).
- Transport field source and destination ports to identify the type of application: ICMP, IGMP, or TCP traffic.

The following table describes NetFlow Lite match parameters. You must configure at least one of the following match parameters for the flow records.

Table 4: Match Parameters

Command	Purpose
match datalink {**ethertype** | **mac** {**destination address input** | **source address input**}}	Specifies a match to datalink or Layer 2 fields. The following command options are available: • **ethertype—**Matches to the ethertype of the packet. • **mac—**Matches the source or destination MAC address from packets at input. **Note** When a datalink flow monitor is assigned to an interface or VLAN, it only creates flows for non-IPv6 or non-IPv4 traffic.

Command	Purpose
match ipv4 {**destination** {**address**} | **protocol** | **source** {**address**} | **tos**}	Specifies a match to the IPv4 fields. The following command options are available: • **destination—**Matches to the IPv4 destination address-based fields. • **protocol—**Matches to the IPv4 protocols. • **source—**Matches to the IPv4 source address based fields. • **tos—**Matches to the IPv4 Type of Service fields.
match ipv6 {**destination** {**address**} | **flow-label** | **protocol** | **source** {**address**} | **traffic-class**}	Specifies a match to the IPv6 fields. The following command options are available: • **destination—**Matches to the IPv6 destination address-based fields. • **flow-label—**Matches to the IPv6 flow-label fields. • **protocol—**Matches to the IPv6 payload protocol fields. • **source—**Matches to the IPv6 source address based fields. • **traffic-class—**Matches to the IPv6 traffic class.
match transport {**destination-port** | **source-port**}	Specifies a match to the Transport Layer fields. The following command options are available: • **destination-port—**Matches to the transport destination port. • **source-port—**Matches to the transport source port.
	Specifies the use of SSID of the wireless network as a key field for a flow record.

NetFlow Lite Collect Parameters

You can collect these key fields in the flow record:

- The total number of bytes, flows or packets sent by the exporter (exporter) or the number of bytes or packets in a 64-bit counter (long).

- The timestamp based on system uptime from the time the first packet was sent or from the time the most recent (last) packet was seen.
- The SNMP index of the input interface. The interface for traffic entering the service module is based on the switch forwarding cache. This field is typically used in conjunction with datalink, IPv4, and IPv6 addresses, and provides the actual first-hop interface for directly connected hosts.
  - A value of 0 means that interface information is not available in the cache.
  - Some NetFlow collectors require this information in the flow record.

The following table describes NetFlow Lite collect parameters.

Table 5: Collect Parameters

Command	Purpose
collect counter {**bytes** {**long** | **permanent** } | **packets** { **long** | **permanent**}}	Collects the counter fields total bytes and total packets.
collect flow {**sampler**}	Collects the flow sampler identifier (ID).
collect interface {**input**}	Collects the fields from the input interface.
collect timestamp sys-uptime {**first** | **last**}	Collects the fields for the time the first packet was seen or the time the most recent packet was last seen (in milliseconds).
collect transport tcp flags	Collects the following transport TCP flags: • **ack—**TCP acknowledgement flag • **cwr—**TCP congestion window reduced flag • **ece—**TCP ECN echo flag • **fin—**TCP finish flag • **psh—**TCP push flag • **rst—**TCP reset flag • **syn—**TCP synchronize flag • **urg—**TCP urgent flag
	Collects the MAC addresses of the access points that the wireless client is associated with.

Flow Exporters

Flow exporters export the data in the flow monitor cache to a remote system, such as a server running NetFlow collector, for analysis and storage. Flow exporters are created as separate entities in the configuration. Flow

exporters are assigned to flow monitors to provide data export capability for the flow monitors. You can create several flow exporters and assign them to one or more flow monitors to provide several export destinations. You can create one flow exporter and apply it to several flow monitors.

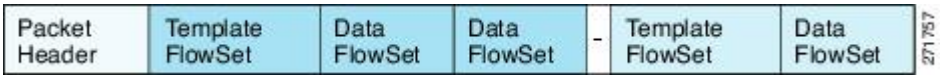
NetFlow Data Export Format Version 9

The basic output of NetFlow is a flow record. Several different formats for flow records have evolved as NetFlow has matured. The most recent evolution of the NetFlow export format is known as Version 9. The distinguishing feature of the NetFlow Version 9 export format is that it is template-based. Templates provide an extensible design to the record format, a feature that should allow future enhancements to NetFlow services without requiring concurrent changes to the basic flow-record format. Using templates provides several key benefits:

- Third-party business partners who produce applications that provide collector or display services for NetFlow do not have to recompile their applications each time a new NetFlow feature is added. Instead, they should be able to use an external data file that documents the known template formats.
- New features can be added to NetFlow quickly without breaking current implementations.
- NetFlow is “future-proofed” against new or developing protocols because the Version 9 format can be adapted to provide support for them.

The Version 9 export format consists of a packet header followed by one or more template flow or data flow sets. A template flow set provides a description of the fields that will be present in future data flow sets. These data flow sets may occur later within the same export packet or in subsequent export packets. Template flow and data flow sets can be intermingled within a single export packet, as illustrated in the figure below.

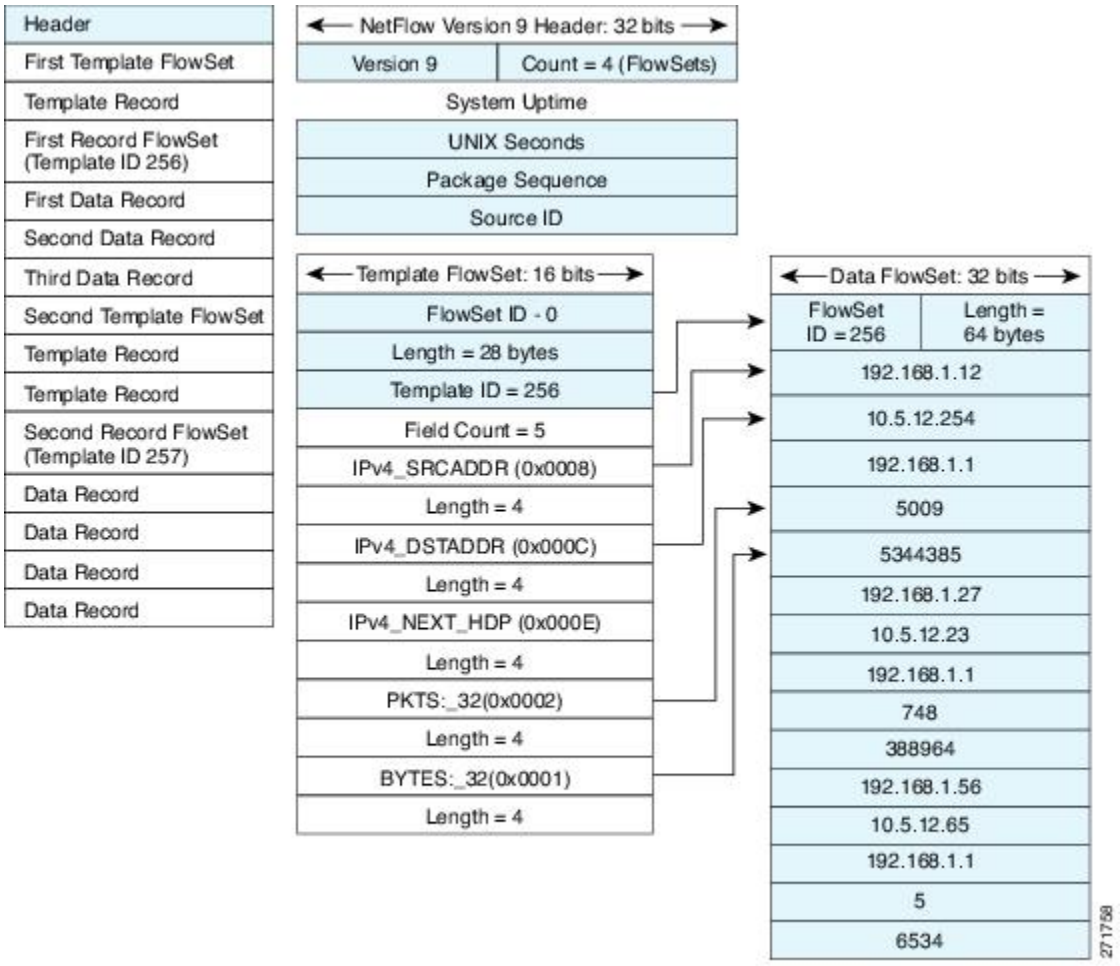
Figure 1: Version 9 Export Packet



NetFlow Version 9 will periodically export the template data so the NetFlow collector will understand what data is to be sent and also export the data flow set for the template. The key advantage to Flexible NetFlow is that the user configures a flow record, which is effectively converted to a Version 9 template and then

forwarded to the collector. The figure below is a detailed example of the NetFlow Version 9 export format, including the header, template flow, and data flow sets.

Figure 2: Detailed Example of the NetFlow Version 9 Export Format



For more information on the Version 9 export format, refer to the white paper titled Cisco IOS NetFlow Version 9 Flow-Record Format, available at this URL: http://www.cisco.com/en/US/tech/tk648/tk362/technologies_white_paper09186a00800a3db9.shtml.

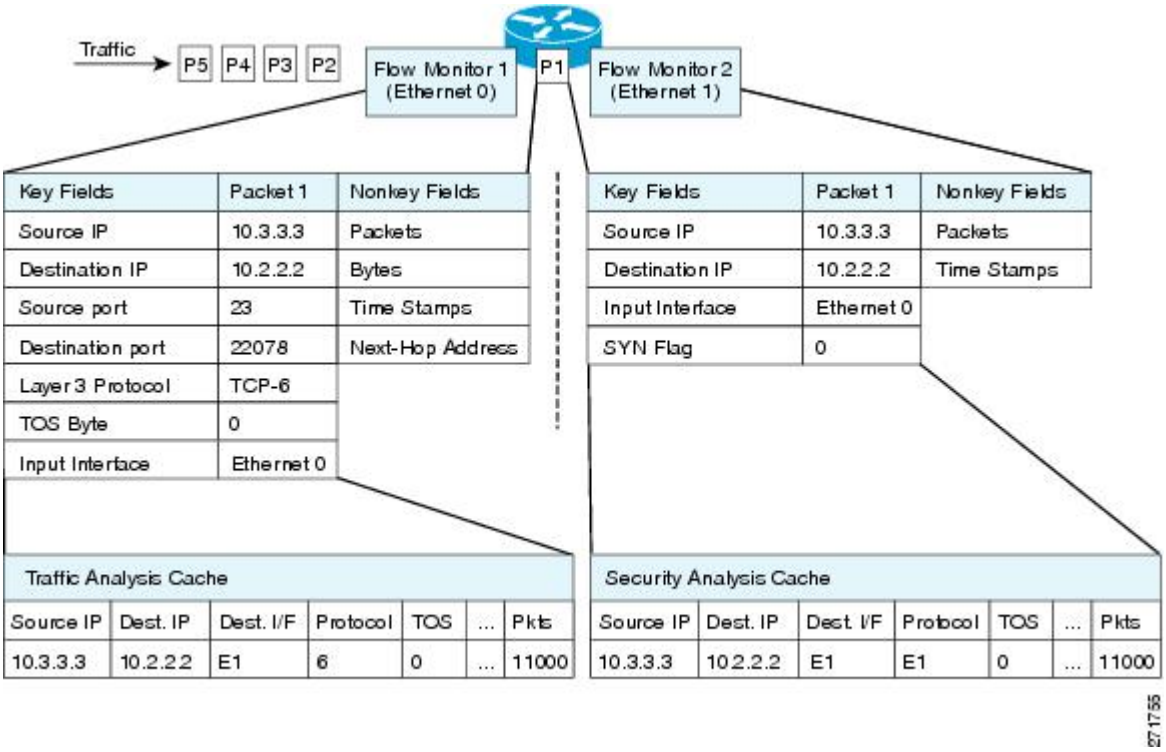
Flow Monitors

Flow monitors are the Flexible NetFlow component that is applied to interfaces to perform network traffic monitoring.

Flow data is collected from the network traffic and added to the flow monitor cache during the monitoring process based on the key and nonkey fields in the flow record.

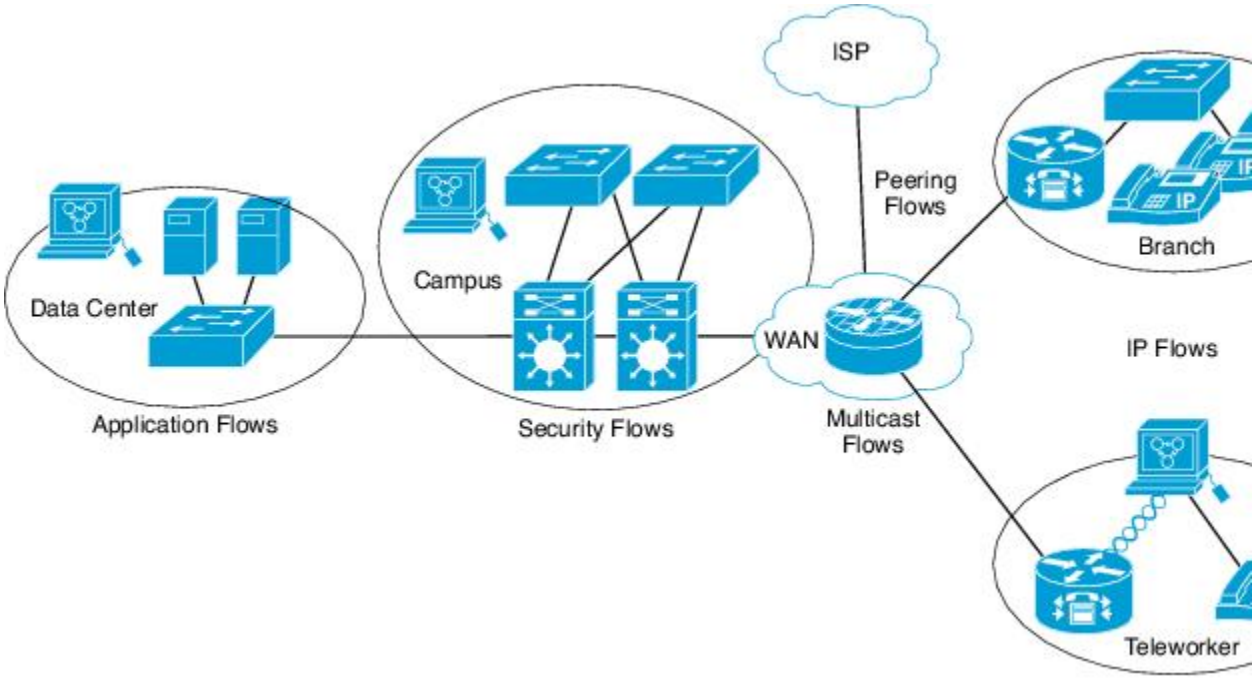
Flexible NetFlow can be used to perform different types of analysis on the same traffic. In the figure below, packet 1 is analyzed using a record designed for standard traffic analysis on the input interface and a record designed for security analysis on the output interface.

Figure 3: Example of Using Two Flow Monitors to Analyze the Same Traffic



The figure below shows a more complex example of how you can apply different types of flow monitors with custom records.

Figure 4: Complex Example of Using Multiple Types of Flow Monitors with Custom Records



Normal

The default cache type is “normal”. In this mode, the entries in the cache are aged out according to the timeout active and timeout inactive settings. When a cache entry is aged out, it is removed from the cache and exported via any exporters configured.

Flow Samplers

Flow samplers are created as separate components in a router’s configuration. Flow samplers are used to reduce the load on the device that is running NetFlow Lite by limiting the number of packets that are selected for analysis.

Samplers use random sampling techniques (modes); that is, a randomly selected sampling position is used each time a sample is taken.

Flow sampling exchanges monitoring accuracy for router performance. When you apply a sampler to a flow monitor, the overhead load on the router of running the flow monitor is reduced because the number of packets that the flow monitor must analyze is reduced. The reduction in the number of packets that are analyzed by the flow monitor causes a corresponding reduction in the accuracy of the information stored in the flow monitor’s cache.

Samplers are combined with flow monitors when they are applied to an interface with the **ip flow monitor** command.

NetFlow Lite and Stacking

The switch supports NetFlow Lite running on a mixed stack configuration, where both Catalyst 2960-X and Catalyst 2960-S switches reside in the same stack. But in such a mixed stack configuration, the master switch must always be a Catalyst 2960-X switch. The Catalyst 2960-S switch must never be the master switch in this type of mixed stack configuration.

Default Settings

The following table lists the NetFlow Lite default settings for the switch.

Table 6: Default NetFlow Lite Settings

Setting	Default
Flow active timeout	1800 seconds **Note** The default value for this setting may be too high for your specific NetFlow Lite configuration. You may want to consider changing it to a lower value of 180 or 300 seconds.
Flow timeout inactive	Enabled, 30 seconds
Flow update timeout	1800 seconds
Default cache size	16640 bits

How to Configure NetFlow Lite

To configure NetFlow Lite, follow these general steps:

1. Create a flow record by specifying keys and non-key fields to the flow.
2. Create an optional flow exporter by specifying the protocol and transport destination port, destination, and other parameters.
3. Create a flow monitor based on the flow record and flow exporter.
4. Create an optional sampler.
5. Apply the flow monitor to a Layer 2 port, Layer 3 port, or VLAN.

Creating a Flow Record

You can create a flow record and add keys to match on and fields to collect in the flow.

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *name*
3. **description** *string*
4. **match** *type*
5. **collect** *type*
6. **end**
7. **show flow record** [**name** *record-name*]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**flow record** *name*
	
	Example:
	
	`Switch(config)# flow record test`
	`Switch(config-flow-record)#`
		Creates a flow record and enters flow record configuration mode.

Step 3	**description** *string* **Example:** `Switch(config-flow-record)# description Ipv4Flow`	(Optional) Describes this flow record as a maximum 63-character string.

Step 4	**match** *type*
	
	Example:
	
	`Switch(config-flow-record)# match ipv4 source address`
	`Switch(config-flow-record)# match ipv4 destination address`
	`Switch(config-flow-record)# match flow direction`
		Specifies a match key. For information about possible match key values, see Flexible NetFlow Match Parameters.

Step 5	**collect** *type* **Example:** `Switch(config-flow-record)# collect counter`	Specifies the collection field. For information about possible collection field values, see Flexible NetFlow Collect Parameters.

	Command or Action	Purpose
	bytes layer2 long Switch(config-flow-record)# **collect counter bytes long** Switch(config-flow-record)# **collect timestamp absolute first** Switch(config-flow-record)# **collect transport tcp flags** Switch(config-flow-record)# collect interface output	**Note** When a flow monitor has the **collect interface output** as the collect field in the flow record, then the output interface is detected based on the destination address in the switch. Hence, for the different flow monitors, the following are required to be configured: • For ipv4 flow monitor, configure "**match ip destination address**" • For ipv6 flow monitor, configure "**match ipv6 destination address**" • For datalink flow monitor, configure "**match datalink mac output**" The **collect interface output** field will return a value of **NULL** when a flow gets created for any of the following addresses: • L3 broadcast • L2 broadcast • L3 Multicast • L2 Multicast • L2 unknown destination.
Step 6	**end** **Example:** Switch(config-flow-record)# **end**	Returns to privileged EXEC mode.
Step 7	**show flow record** [**name** *record-name*] **Example:** Switch **show flow record test**	(Optional) Displays information about NetFlow flow records.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to Do Next

Define an optional flow exporter by specifying the export format, protocol, destination, and other parameters.

Related Topics

Flow Records, on page 16
Example: Configuring a Flow, on page 38

Creating a Flow Exporter

You can create a flow export to define the export parameters for a flow.

Note Each flow exporter supports only one destination. If you want to export the data to multiple destinations, you must configure multiple flow exporters and assign them to the flow monitor.

You can export to a destination using IPv4 address.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** *name*
3. **description** *string*
4. **destination** {*ipv4-address*} [**vrf** *vrf-name*]
5. **dscp** *value*
6. **source** { *source type* }
7. **transport udp** *number*
8. **ttl** *seconds*
9. **export-protocol** {**netflow-v9**}
10. **end**
11. **show flow exporter** [**name** *record-name*]
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**flow exporter** *name* **Example:** `Switch(config)# flow exporter ExportTest`	Creates a flow exporter and enters flow exporter configuration mode.
Step 3	**description** *string* **Example:** `Switch(config-flow-exporter)# description ExportV9`	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	**destination** {*ipv4-address*} [**vrf** *vrf-name*] **Example:** `Switch(config-flow-exporter)# destination 192.0.2.1 (IPv4 destination)`	Sets the IPv4 destination address or hostname for this exporter.
Step 5	**dscp** *value* **Example:** `Switch(config-flow-exporter)# dscp 0`	(Optional) Specifies the differentiated services codepoint value. The range is from 0 to 63. The default is 0.
Step 6	**source** { *source type* } **Example:** `Switch(config-flow-exporter)# source gigabitEthernet1/0/1`	(Optional) Specifies the interface to use to reach the NetFlow collector at the configured destination. The following interfaces can be configured as source:
Step 7	**transport udp** *number* **Example:** `Switch(config-flow-exporter)# transport udp 200`	(Optional) Specifies the UDP port to use to reach the NetFlow collector. The range is from 1 to 65536
Step 8	**ttl** *seconds* **Example:** `Switch(config-flow-exporter)# ttl 210`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. The range is from 1 to 255 seconds. The default is 255.

	Command or Action	Purpose
Step 9	**export-protocol** {**netflow-v9**} **Example:** `Switch(config-flow-exporter)# export-protocol netflow-v9`	Specifies the version of the NetFlow export protocol used by the exporter.
Step 10	**end** **Example:** `Switch(config-flow-record)# end`	Returns to privileged EXEC mode.
Step 11	**show flow exporter** [**name** *record-name*] **Example:** `Switch show flow exporter ExportTest`	(Optional) Displays information about NetFlow flow exporters.
Step 12	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Define a flow monitor based on the flow record and flow exporter.

Related Topics

Exporters
Example: Configuring a Flow, on page 38

Creating a Flow Monitor

You can create a flow monitor and associate it with a flow record and a flow exporter.

SUMMARY STEPS

1. **configure terminal**
2. **flow monitor** *name*
3. **description** *string*
4. **exporter** *name*
5. **record** *name*
6. **cache** { **timeout** {**active** | **inactive**} *seconds* | **type normal** }
7. **end**
8. **show flow monitor** [**name** *record-name*]
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**flow monitor** *name*
	
	Example:
	`Switch(config)# flow monitor MonitorTest`
	`Switch (config-flow-monitor)#`
		Creates a flow monitor and enters flow monitor configuration mode.

Step 3	**description** *string* **Example:** `Switch(config-flow-monitor)# description Ipv4Monitor`	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	**exporter** *name* **Example:** `Switch(config-flow-monitor)# exporter ExportTest`	Associates a flow exporter with this flow monitor.
Step 5	**record** *name* **Example:** `Switch(config-flow-monitor)# record test`	Associates a flow record with the specified flow monitor.
Step 6	**cache** { **timeout** {**active**	**inactive**} *seconds*	**type normal** }	Associates a flow cache with the specified flow monitor.

	Command or Action	Purpose
	Example: `Switch(config-flow-monitor)# cache timeout active 15000`	
Step 7	**end** **Example:** `Switch(config-flow-monitor)# end`	Returns to privileged EXEC mode.
Step 8	**show flow monitor** [**name** *record-name*] **Example:** `Switch show flow monitor name MonitorTest`	(Optional) Displays information about NetFlow flow monitors.
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Apply the flow monitor to a Layer 2 interface, Layer 3 interface, or VLAN.

Related Topics

Monitors
Example: Configuring a Flow, on page 38

Creating a Sampler

You can create a sampler to define the NetFlow sampling rate for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **sampler** *name*
3. **description** *string*
4. **mode** { **deterministic** { *m* - *n* } | **random** { *m* - *n* }}
5. **end**
6. **show sampler** [*name*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**sampler** *name*
	
	Example:
	
	`Switch(config)# sampler SampleTest`
	`Switch(config-flow-sampler)#`
		Creates a sampler and enters flow sampler configuration mode.

Step 3	**description** *string* **Example:** `Switch(config-flow-sampler)# description samples`	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	**mode** { **deterministic** { *m* - *n* } | **random** { *m* - *n* }} **Example:** `Switch(config-flow-sampler)# mode random 1 out-of 1022`	Defines the random sample mode. You can configure either a random or deterministic sampler to an interface. Select *m* packets out of an *n* packet window. The window size to select packets from ranges from 32 to 1022. Note the following when configuring a sampler to an interface: • When you attach a monitor using deterministic sampler (for example, s1), every attachment with same sampler s1 uses one new free sampler from the switch (hardware) out of 4 available samplers. Therefore, beyond 4 attachments, you are not allowed to attach a monitor with any sampler. • In contrast, when you attach a monitor using random sampler (for example-again, s1), only the first attachment uses a new sampler from the switch (hardware). The rest of all attachments using the same sampler s1, share the same sampler.

	Command or Action	Purpose
		Due to this behavior, when using a deterministic sampler, you can always make sure the correct number of flows are sampled by comparing the sampling rate and what the switch sends. If the same random sampler is used with multiple interfaces, flows from an interface can always be sampled, and the flows from other interfaces could be always skipped.
Step 5	**end** **Example:** `Switch(config-flow-sampler)# end`	Returns to privileged EXEC mode.
Step 6	**show sampler** [*name*] **Example:** `Switch show sample SampleTest`	(Optional) Displays information about NetFlow samplers.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Apply the flow monitor to a source interface or a VLAN.

Applying a Flow to an Interface

You can apply a flow monitor and an optional sampler to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. {**ip flow monitor** | **ipv6 flow monitor**}*name* [|**sampler** *name*] { **input** |**output** }
4. **end**
5. **show flow interface** [*interface-type number*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *type* **Example:** `Switch(config)# interface GigabitEthernet1/0/1`	Enters interface configuration mode and configures an interface. Command parameters for the interface configuration include: You cannot attach a NetFlow monitor to a port channel interface. If both service module interfaces are part of an EtherChannel, you should attach the monitor to both physical interfaces.
Step 3	{**ip flow monitor** | **ipv6 flow monitor**}*name* [|**sampler** *name*] { **input** |**output** } **Example:** `Switch(config-if)# ip flow monitor MonitorTest input`	Associate an IPv4 or an IPv6 flow monitor, and an optional sampler to the interface for input or output packets. To monitor datalink L2 traffic flows, you would use **datalink flow monitor** *name* **sampler** *sampler-name* {**input**} interface command. This specific command associates a datalink L2 flow monitor and required sampler to the interface for input packets. When a datalink flow monitor is assigned to an interface or VLAN record, it only creates flows for non-IPv6 or non-IPv4 traffic. **Note** Whenever you assign a flow monitor to an interface, you must configure a sampler. If the sampler is missing, you will receive an error message.
Step 4	**end** **Example:** `Switch(config-flow-monitor)# end`	Returns to privileged EXEC mode.
Step 5	**show flow interface** [*interface-type number*] **Example:** `Switch# show flow interface`	(Optional) Displays information about NetFlow on an interface.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Bridged NetFlow on a VLAN

You can apply a flow monitor and an optional sampler to a VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** [**configuration**] *vlan-id*
3. **interface** {**vlan**} *vlan-id*
4. **ip flow monitor** *monitor name* [**sampler** *sampler name*] {**input** |**output**}
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**vlan** [**configuration**] *vlan-id*
	
	Example:
	
	`Switch(config)# vlan configuration 30`
	`Switch(config-vlan-config)#`
		Enters VLAN or VLAN configuration mode.

Step 3	**interface** {**vlan**} *vlan-id* **Example:** `Switch(config)# interface vlan 30`	Specifies the SVI for the configuration.
Step 4	**ip flow monitor** *monitor name* [**sampler** *sampler name*] {**input** |**output**} **Example:** `Switch(config-vlan-config)# ip flow monitor MonitorTest input`	Associates a flow monitor and an optional sampler to the VLAN for input or output packets.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 NetFlow

You can define Layer 2 keys in NetFlow Lite records that you can use to capture flows in Layer 2 interfaces.

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *name*
3. **match datalink** { **ethertype** | **mac** { **destination** { **address input** } | **source** { **address input** } } }
4. **match** { **ipv4** {**destination** | **protocol** | **source** | **tos** } | **ipv6** {**destination** | **flow-label**| **protocol**| **source**| **traffic-class** } | **transport** {**destination-port** | **source-port**} }
5. **end**
6. **show flow record** [*name*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**flow record** *name*
	
	Example:
	
	`Switch(config)# flow record L2_record`
	`Switch(config-flow-record)#`
		Enters flow record configuration mode.

Step 3	**match datalink** { **ethertype**	**mac** { **destination** { **address input** }	**source** { **address input** } } }
			
			Example:
			
			`Switch(config-flow-record)# match datalink mac source address input`
			`Switch(config-flow-record)# match datalink mac destination address input`
				Specifies the Layer 2 attribute as a key. In this example, the keys are the source and destination MAC addresses from the packet at input.
				
				Note When a datalink flow monitor is assigned to an interface or VLAN record, it only creates flows for non-IPv4 or non-IPv6 traffic.

Step 4	**match** { **ipv4** {**destination**	**protocol**	**source**	**tos** }	**ipv6** {**destination**	**flow-label**| **protocol**| **source**| **traffic-class** }	**transport** {**destination-port**	**source-port**} }	Specifies additional Layer 2 attributes as a key. In this example, the keys are IPv4 protocol and ToS.

	Command or Action	Purpose

	Example:
	`Switch(config-flow-record)# match ipv4 protocol`
	`Switch(config-flow-record)# match ipv4 tos`	

Step 5	**end** **Example:** `Switch(config-flow-record)# end`	Returns to privileged EXEC mode.
Step 6	**show flow record** [*name*] **Example:** `Switch# show flow record`	(Optional) Displays information about NetFlow on an interface.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Flexible NetFlow

The commands in the following table can be used to monitor Flexible NetFlow.

Table 7: Flexible NetFlow Monitoring Commands

Command	Purpose
show flow exporter [**broker** | **export-ids** | **name** | *name* | **statistics** | **templates**]	Displays information about NetFlow flow exporters and statistics.
show flow exporter [**name** *exporter-name*]	Displays information about NetFlow flow exporters and statistics.
show flow interface	Displays information about NetFlow interfaces.
show flow monitor [**name** *exporter-name*]	Displays information about NetFlow flow monitors and statistics.
show flow monitor statistics	Displays the statistics for the flow monitor

Command	Purpose
show flow monitor cache format {**table** | **record** | **csv**}	Displays the contents of the cache for the flow monitor, in the format specified.
show flow record [**name** *record-name*]	Displays information about NetFlow flow records.
show flow ssid	Displays NetFlow monitor installation status for a WLAN.
show sampler [**broker** | **name** | *name*]	Displays information about NetFlow samplers.
show wlan *wlan-name*	Displays the WLAN configured on the device.

Configuration Examples for NetFlow Lite

Example: Configuring a Flow

Note When configuring a flow, you need to have the protocol, source port, destination port, first and last timestamps, and packet and bytes counters defined in the flow record. Otherwise, you will get the following error message: "Warning: Cannot set protocol distribution with this Flow Record. Require protocol, source and destination ports, first and last timestamps and packet and bytes counters."

This example shows how to create a flow and apply it to an interface:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)# flow exporter export1
Switch(config-flow-exporter)# destination 10.0.101.254
Switch(config-flow-exporter)# transport udp 2055
Switch(config-flow-exporter)# template data timeout 60
Switch(config-flow-exporter)# exit
Switch(config)# flow record record1
Switch(config-flow-record)# match ipv4 source address
Switch(config-flow-record)# match ipv4 destination address
Switch(config-flow-record)# match ipv4 protocol
Switch(config-flow-record)# match transport source-port
Switch(config-flow-record)# match transport destination-port
Switch(config-flow-record)# collect counter bytes long
Switch(config-flow-record)# collect counter packets long
Switch(config-flow-record)# collect timestamp sys-uptime first
Switch(config-flow-record)# collect timestamp sys-uptime last
Switch(config-flow-record)# exit
Switch(config)# sampler SampleTest
Switch(config-sampler)# mode random 1 out-of 100
Switch(config-sampler)# exit
Switch(config)# flow monitor monitor1
Switch(config-flow-monitor)# cache timeout active 300
Switch(config-flow-monitor)# cache timeout inactive 120
Switch(config-flow-monitor)# record record1
Switch(config-flow-monitor)# exporter export1
```

```
Switch(config-flow-monitor)# exit
Switch(config)# interface GigabitEthernet1/0/1
Switch(config-if)# ip flow monitor monitor1 sampler SampleTest input
Switch(config-if)# end
```

Related Topics

Creating a Flow Record, on page 24
Flow Records, on page 16
Creating a Flow Exporter, on page 27
Exporters
Creating a Flow Monitor, on page 29
Monitors
Creating a Sampler
Samplers

Additional References

Related Documents

Related Topic	Document Title
Flexible NetFlow CLI Commands	*Flexible NetFlow Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
RFC 3954	Cisco Systems NetFlow Services Export Version 9

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Flexible NetFlow

Release	Modification
Cisco IOS 15.0(2)EX	This feature was introduced.

INDEX