Configuring SNMP and using the NetFlow MIB to Monitor NetFlow Data

Last Updated: April 1, 2012

NetFlow is a technology that provides highly granular per-flow statistics on traffic in a Cisco router. The NetFlow MIB feature provides MIB objects to allow users to configure NetFlow and to monitor flow cache information, the current NetFlow configuration, and statistics.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. 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 document.

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

Prerequisites for Configuring SNMP and the NetFlow MIB to Monitor NetFlow Data

Before you enable NetFlow you must:

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

- Configure the router for IP routing
- Ensure that one of the following is enabled on your router, and on the interfaces that you want to configure NetFlow on: Cisco Express Forwarding (CEF), distributed CEF, or fast switching
- Understand the resources required on your router because NetFlow consumes additional memory and CPU resources
- Configure SNMP on the router on which the NetFlow MIB feature is to be used. Refer to the Configuring the Router to use SNMP, page 5 for more information. For more information on configuring an SNMP server, refer to the Configuring SNMP Support in the *Cisco IOS Network Management Configuration Guide* .

Restrictions for Configuring SNMP and the NetFlow MIB to Monitor NetFlow Data

Cisco IOS Releases 12.2(14)S, 12.0(22)S, or 12.2(15)T

If your router is running a version of Cisco IOS prior to releases 12.2(14)S, 12.0(22)S, or 12.2(15)T the **ip route-cache flow** command is used to enable NetFlow on an interface.

If your router is running Cisco IOS release 12.2(14)S, 12.0(22)S, 12.2(15)T, or later the **ip flow ingress** command is used to enable NetFlow on an interface.

Information About Configuring SNMP and the NetFlow MIB to Monitor NetFlow Data

NetFlow MIB Feature Benefits

NetFlow is a technology that collects traffic flow statistics on routing devices. NetFlow has been used for a variety of applications, including traffic engineering, usage-based billing, and denial of service (DoS) attack monitoring.

The NetFlow MIB feature is useful for obtaining IP flow information from a Cisco router when a NetFlow export operation is not possible. NetFlow exporting does not have to be enabled for the NetFlow MIB feature to be used. The NetFlow MIB feature can be implemented instantaneously at any point in the network to obtain flow information.

With the NetFlow MIB feature, system information that is stored in the flow cache can be accessed in real time by utilizing a MIB implementation based on SNMP. This information is accessed using **get** and **set** commands entered on the network management system (NMS) workstation for which SNMP has been implemented. The NMS workstation is also known as the SNMP manager.

NetFlow MIB Overview

The Netflow MIB provides a simple and easy method to configure NetFlow, NetFlow aggregation caches, and NetFlow Data Export. You use the snmpget and snmpwalk tools to get NetFlow cache information and current NetFlow configuration information. The NetFlow MIB feature enables medium to small size enterprises to take advantage of NetFlow technology over SNMP at a reduced infrastructure cost. The MIB is created to provide Netflow information in these areas:

- Cache information and configuration.
- Export information and configuration.
- Export Statistics.
- Protocol Statistics.
- Version 9 Export Template information.
- Top Flows information.

- Terminology Used, page 3

Terminology Used

Flow

A flow is defined as an unidirectional sequence of packets between a given source and destination endpoints. Network flows are highly granular; flow endpoints are identified both by IP address as well as by transport layer application port numbers. NetFlow also utilizes the IP Protocol type, Type of Service (ToS) and the input interface identifier to uniquely identify flows.

Exporter

A device (for example, a router) with NetFlow services enabled. The exporter monitors packets entering an observation point and creates flows out of these packets. The information from these flows are exported in the form of Flow Records to the collector. You can configure NetFlow data export using the NetFlow MIB.

Flow Record

A Flow Record provides information about an IP Flow that exists on the Exporter. The Flow Records are commonly referred to as NetFlow Services data or NetFlow data.

Collector

The NetFlow Collector receives Flow Records from one or more Exporters. It processes the received export packet, i.e. parses, stores the Flow Record information. The flow records may be optionally aggregated before storing into the hard disk.

Template

NetFlow Version 9 Export format is template based. Version 9 record format consists of a packet header followed by at least one or more template or data FlowSets. A template FlowSet (collection of one or more template) provides a description of the fields that will be present in future data FlowSets. 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.

One additional record type is also a part of Version 9 specification: an options template. Rather than supplying information about IP flows, options are used to supply meta-data about the NetFlow process itself.

Top Flows

This feature provides a mechanism which allows the top N flows in the NetFlow cache to be viewed in real time.

Criteria can be set to limit the feature to particular flows of interest, which can aid in DoS detection.

Only the number of flows (TopN) and the sort criteria (SortBy) need be set.

Top Flows is not intended as a mechanism for exporting the entire netflow cache.

For more information on the Top Flows and the NetFlow MIB refer to the Configuring NetFlow Top Talkers using Cisco IOS CLI Commands or SNMP Commands.

Egress flows

This feature analyzes traffic that is being forwarded by the router. This feature is often referred to as Egress NetFlow.

Using SNMP and MIBs to Extract NetFlow Information

SNMP has historically been used to collect network information. SNMP permits retrieval of critical information from network elements such as routers, switches, and workstations. The NetFlow MIB feature uses SNMP to configure NetFlow and to gather NetFlow statistics.

The NetFlow MIB feature allows NetFlow statistics and other NetFlow data for the managed devices on your system to be retrieved by SNMP. You can specify retrieval of NetFlow information from a managed device (for example, a router) either by entering commands on that managed device or by entering SNMP commands from the NMS workstation to configure the router via the MIB. If the NetFlow information is configured from the NMS workstation, no access to the router is required and all configuration can be performed via SNMP. The NetFlow MIB request for information is sent from an NMS workstation via SNMP to the router and is retrieved from the router. This information can then be stored or viewed, thus allowing NetFlow information to be easily accessed and transported across a multi-vendor programming environment.

Objects That are Used by the NetFlow MIB

The NetFlow MIB feature defines managed objects that enable a network administrator to remotely monitor the following NetFlow information:

- Flow cache configuration information
- NetFlow export information
- General NetFlow statistics

How to Configure SNMP and use the NetFlow MIB to Monitor NetFlow Data

Note Some of the tasks in this section include examples of the SNMP CLI syntax used to set configuration parameters on the router, and to read values from MIB objects on the router. These SNMP CLI syntax examples are taken from a Linux workstation using public domain SNMP tools. The SNMP CLI syntax for your workstation might be different. Refer to the documentation that was provided with your SNMP tools for the correct syntax for your network management workstation.

Configuring the Router to use SNMP

Before the NetFlow MIB feature can be used, the router must be configured to support SNMP. To enable SNMP on the router, perform this task.

Note The SNMP community read-only (RO) string for the examples is **public**. The SNMP community read-write (RW) string for the examples is **private**. You should use more complex strings for these values in your configurations.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server community** *string* **ro**
4. **snmp-server community** *string* **rw**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	(Required) Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	(Required) Enters global configuration mode.
Step 3	**snmp-server community** *string* **ro** **Example:** `Router(config)# snmp-server community public ro`	(Required) Sets up the community access string to permit access to SNMP. • The *string* argument is a community string that consists of from 1 to 32 alphanumeric characters and functions much like a password, permitting access to the SNMP protocol. Blank spaces are not permitted in the community string. • The **ro** keyword specifies read-only access. SNMP management stations using this string can retrieve MIB objects.
Step 4	**snmp-server community** *string* **rw** **Example:** `Router(config)# snmp-server community private rw`	(Required) Sets up the community access string to permit access to SNMP. • The *string* argument is a community string that consists of from 1 to 32 alphanumeric characters and functions much like a password, permitting access to the SNMP protocol. Blank spaces are not permitted in the community string. • The **rw** keyword specifies read-write access. SNMP management stations using this string can retrieve and modify MIB objects. **Note** The *string* argument must be different from the read-only *string* argument specified in the preceding step (Step 3).
Step 5	**end** **Example:** `Router(config)# end`	(Required) Exits the current configuration mode and returns to privileged EXEC mode.

Configuring Options for the Main Cache

This optional task describes the procedure for modifying the parameters for the NetFlow main cache. Perform the steps in this optional task using either the router CLI commands or the SNMP commands to modify the parameters for the NetFlow main cache.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip flow-cache entries** *number*
4. **ip flow-cache timeout active** *minutes*
5. **ip flow-cache timeout inactive** *seconds*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	(Required) Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	(Required) Enters global configuration mode.
Step 3	**ip flow-cache entries** *number* **Example:** `Router(config)# ip flow-cache entries 4000`	(Optional) Specifies the maximum number of entries to be captured for the main flow cache. **Note** The valid range for the *number* argument is from 1024 to 524288 entries.
Step 4	**ip flow-cache timeout active** *minutes* **Example:** `Router(config)# ip flow-cache timeout active 30`	(Optional) Configures operational parameters for the main cache. • The **timeout** keyword dissolves the session in the cache. • The **active** *minutes* keyword-argument pair is the number of minutes that an entry is active. The range is from 1 to 60 minutes. The default is 30 minutes.
Step 5	**ip flow-cache timeout inactive** *seconds* **Example:** `Router(config)# ip flow-cache timeout inactive 100`	(Optional) Configures operational parameters for the main cache. • The **timeout** keyword dissolves the session in the main cache. • The **inactive** *seconds*keyword-argument pair is the number of seconds that an inactive entry will stay in the main cache before it times out. The range is from 10 to 600 seconds. The default is 15 seconds.

	Command or Action	Purpose
Step 6	**end** **Example:** `Router(config)# end`	(Required) Exits the current configuration mode and returns to privileged EXEC mode.

Configuring Options for the Main Cache

SUMMARY STEPS

1. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCICacheEntries.type unsigned** *number*
2. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCIActiveTimeOut.type unsigned** *number*
3. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] c**nfCIInactiveTimeOut.type unsigned** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCICacheEntries.type unsigned** *number* **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.62 cnfCICacheEntries.0 unsigned 4000`	(Optional) Defines the maximum number of entries to be captured for the main flow cache. • The value for the *type* argument in **cnfCICacheEntries**.*type* **unsigned** *number* is 0 for the main cache. • The value for the *number* argument in **cnfCICacheEntries**.*typenumber* is the maximum number of cache entries. **Note** The valid range for the *number* argument is from 1024 to 524288 entries.
Step 2	**snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCIActiveTimeOut.type unsigned** *number* **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.62 cnfCIActiveTimeOut.0 unsigned 60`	(Optional) Specifies the number of seconds that an active flow remains in the main cache before it times out. • The value for the *type* argument in **cnfCIActiveTimeout**.*type* **unsigned** *number* is 0 for the main cache. • The value for the *number* argument in **cnfCIActiveTimeout**.*type* **unsigned** *number* is the number of seconds that an active flow remains in the cache before it times out. **Note** The range for the *number* argument is from 1 to 60 minutes. The default is 30 minutes.

Command or Action	Purpose
Step 3 **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] c**nfCIInactiveTimeOut.type unsigned** *number* **Example**: `workstation% snmpset -c private -m all -v2c 10.4.9.62 cnfCIInactiveTimeOut.0 unsigned 30`	(Optional) Specifies the number of seconds that an inactive flow remains in the main cache before it times out. • The value for the *type* argument in **cnfCIInactiveTimeout**.*type* **unsigned** *number* is 0 for the main cache. • The value for the *number* argument in **cnfCIInactiveTimeout**.*type* **unsigned** *number* is the number of seconds that an inactive flow remains in the main cache before it times out. **Note** The range for the *number* argument is from 10 to 600 seconds. The default is 15 seconds.

Identifying the Interface Number to use for Enabling NetFlow with SNMP

Before you can use SNMP to enable NetFlow on an interface, you must identify the correct SNMP interface number on the router. To identify the interface number for the interface that you want to enable NetFlow on, perform the steps in this task.

SUMMARY STEPS

1. **enable**
2. **show snmp mib ifmib ifindex** *type number*

DETAILED STEPS

Step 1 **enable**
Enters privileged EXEC mode. Enter the password if prompted.

Example:

```
Router> enable
```

Step 2 **show snmp mib ifmib ifindex** *type number*
Displays the SNMP interface number for the interface specified.

Example:

```
Router# show snmp mib ifmib ifindex fastethernet 0/0
Ethernet0/0: Ifindex = 1
```

Configuring NetFlow on an Interface

Perform the task using either the router CLI commands or the SNMP commands to enable NetFlow on the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip flow** {**ingress** | **egress**}
5. **exit**
6. Repeat Steps 3 through 5 to enable NetFlow on other interfaces.
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	(Required) Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	(Required) Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface fastethernet0/0`	(Required) Specifies the interface that you want to enable NetFlow on and enters interface configuration mode.

Step 4	**ip flow** {**ingress**	**egress**}
		
		Example:
		`Router(config-if)# ip flow ingress`
		
		Example:
		`and/or`
		
		Example:
		`Router(config-if)# ip flow egress`
			(Required) Enables NetFlow on the interface.
			• **ingress** --captures traffic that is being received by the interface
			• **egress** --captures traffic that is being transmitted by the interface.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode and returns to global configuration mode. **Note** You only need to use this command if you want to enable NetFlow on another interface.
Step 6	Repeat Steps 3 through 5 to enable NetFlow on other interfaces.	(Optional) --
Step 7	**end** **Example:** `Router(config-if)# end`	(Required) Exits the current configuration mode and returns to privileged EXEC mode.

Configuring NetFlow on an Interface

SUMMARY STEPS

1. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] cnfCINetflowEnable.*interface-number* integer [0 | 1 | 2 | 3]
2. Repeat Step 1 to enable NetFlow on other interfaces

DETAILED STEPS

	Command or Action	Purpose
Step 1	**snmpset -c private -m all -v2c** [*ip-address*	*hostname*] cnfCINetflowEnable.*interface-number* integer [0	1	2	3] **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.62 cnfCINetflowEnable.1 integer 1`	(Required) Configures NetFlow for an interface. **Note** The value for the *interface-number* argument is found by entering the router CLI command **show snmp mib ifmib ifindex** on the router in privileged EXEC mode. The values for the *direction* argument are: • 0--Disable NetFlow • 1--Enable Ingress NetFlow • 2--Enable Egress NetFlow • 3--Enable Ingress and Egress NetFlow
Step 2	Repeat Step 1 to enable NetFlow on other interfaces	(Optional) --

Configuring the Destination-Prefix Aggregation Cache

This task describes the procedure for modifying the parameters for aggregation caches. The **destination-prefix** is used in this task. With the exception of specifying the aggregation cache that you want to modify, the steps are the same for modifying these parameters for the other aggregation caches.

Perform this task using either the router CLI commands or the SNMP commands to modify configuration parameters for an aggregation cache.

You must enable NetFlow on at least one interface before configuring a NetFlow aggregation cache.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip flow-aggregation cache destination-prefix**
4. **cache entries** *number*
5. **cache timeout active** *minutes*
6. **cache timeout inactive** *seconds*
7. enable
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	(Required) Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	(Required) Enters global configuration mode.
Step 3	**ip flow-aggregation cache destination-prefix** **Example:** `Router(config)# ip flow-aggregation cache destination-prefix`	(Required) Enters aggregation cache configuration mode for the **destination-prefix**aggregation cache. • The **destination-prefix**keyword is equivalent to the *type* argument of 4 in Step 2 of the SNMP commands. **Note** For information on other keywords for this command, see the *Cisco IOS NetFlow Command Reference* .
Step 4	**cache entries** *number* **Example:** `Router(config-flow-cache)# cache entries 4000`	(Optional) Defines the number of entries that are allowed in the aggregation flow cache.

	Command or Action	Purpose
Step 5	**cache timeout active** *minutes* **Example:** `Router(config)# cache timeout active 30`	(Optional) Specifies the number of minutes that an active flow remains in the cache before it times out. **Note** The range is from 1 to 60 minutes. The default is 30 minutes.

Step 6	**cache timeout inactive** *seconds*
	
	Example:
	`Router(config-flow-cache)`
	`# cache timeout inactive 100`
		(Optional) Specifies the number of seconds that an inactive flow remains in the cache before it times out.
		Note The range is from 10 to 600 seconds. The default is 15 seconds.

Step 7	enable
	
	Example:
	`Router(config-flow-cache)`
	`# enable`
		(Required) Activates the **destination-prefix**aggregation cache.

Step 8	**end** **Example:** `Router(config-if)# end`	(Required) Exits the current configuration mode and returns to privileged EXEC mode.

Configuring the Destination-Prefix Aggregation Cache

SUMMARY STEPS

1. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCICacheEnable**.*type* **integer** *truth-value*
2. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCICacheEntries.** *type* **unsigned** *number*
3. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCIActiveTimeOut.** *type* **unsigned** *number*
4. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] c**nfCIInactiveTimeOut.** *type* **unsigned** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCICacheEnable**.*type* **integer** *truth-value* **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCICacheEnable.4 integer 1`	(Required) Enables the aggregation cache. • Values for the *type* argument are: ◦ Main--0 ◦ AS--1 ◦ Protocol Port--2 ◦ Source Prefix--3 ◦ Destination Prefix--4 ◦ prefix--5 ◦ Destination Only--6 ◦ Source Destination--7 ◦ Full Flow--8 ◦ AS ToS--9 ◦ Protocol Port ToS--10 ◦ Source Prefix ToS--11 ◦ Destination Prefix Tos--12 ◦ Prefix Tos--13 ◦ Prefix Port--14 ◦ BGP Nexthop Tos--15 • Values for *truth-value* in **cnfCICacheEnable**.*type* **integer** *truth-value*are: ◦ 1--enable the aggregation cache ◦ 2--disable the aggregation cache
Step 2	**snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCICacheEntries.** *type* **unsigned** *number* **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.62 cnfCICacheEntries.4 unsigned 4000`	(Optional) Defines the maximum number of entries to be captured for the aggregation flow cache. • The value for the *type* argument in **cnfCICacheEntries**.*type* **unsigned** *number* is 4 for the **destination-prefix** cache. • The value for the *number* argument in **cnfCICacheEntries**.*type* **unsigned** *number* is the maximum number of cache entries. **Note** The valid range for the *number* argument is from 1024 to 524288 entries.

Command or Action	Purpose
Step 3 **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfCIActiveTimeOut.** *type* **unsigned** *number* **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCIActiveTimeOut.4 unsigned 60`	(Optional) Specifies the number of seconds that an active flow remains in the cache before it times out. • The value for the *type* argument in **cnfCIActiveTimeout**.*type* **unsigned** *number* is 4 for the **destination-prefix** cache. • The value for the *number* argument in **cnfCIActiveTimeout**.*type* **unsigned** *number* is the number of seconds that an active flow remains in the cache before it times out. **Note** The range for the *number* argument is from 1 to 60 minutes. The default is 30 minutes.
Step 4 **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] c**cnfCIInactiveTimeOut.** *type* **unsigned** *number* **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCIInactiveTimeOut.4 unsigned 30`	(Optional) Specifies the number of seconds that an inactive flow remains in the cache before it times out. • The value for the *type* argument in **cnfCIInactiveTimeout**.*type* **unsigned** *number* is 4 for the **destination-prefix** cache. • The value for the *number* argument in **cnfCIInactiveTimeout**.*type* **unsigned** *number* is the number of seconds that an inactive flow remains in the cache before it times out. **Note** The range for the *number* argument is from 10 to 600 seconds. The default is 15 seconds.

Configuring NetFlow Export from the Main NetFlow Cache using the Version 9 Export Format

The following example shows how to configure the router to export statistics from the NetFlow main cache (0), including peer autonomous system and BGP-related information using export Version 9.

Perform this task using either the router CLI commands or the SNMP commands to configure the router to export statistics from the main cache using the Version 9.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip flow-export version 9 [origin-as | peer-as] [bgp-nexthop]**
4. **ip flow-export destination** {*ip-address* | *hostname*} *udp-port*}
5. Repeat Step 4 to add a second NetFlow collector
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	(Required) Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	(Required) Enters global configuration mode.
Step 3	**ip flow-export version 9 [origin-as | peer-as] [bgp-nexthop]** **Example:** `Router(config)# ip flow-export version 9 peer-as bgp-nexthop`	(Required) Enables the export of information in NetFlow cache entries. • The **version 9** keyword specifies that the export packet uses the Version 9 format. • The **origin-as** keyword specifies that export statistics include the originating AS for the source and destination. • The **peer-as** keyword specifies that export statistics include the peer AS for the source and destination. • The **bgp-nexthop** keyword specifies that export statistics include BGP next hop-related information. **Caution** Entering this command on a Cisco 12000 Series Internet Router causes packet forwarding to stop for a few seconds while NetFlow reloads the route processor and line card CEF tables. To avoid interruption of service to a live network, apply this command during a change window, or include it in the startup-config file to be executed during a router reboot.
Step 4	**ip flow-export destination** {*ip-address* | *hostname*} *udp-port*} **Example:** `Router(config)# ip flow-export destination 10.0.19.2 999`	(Required) Specifies the IP address, or hostname of the NetFlow collector, and the UDP port the NetFlow collector is listening on.
Step 5	Repeat Step 4 to add a second NetFlow collector	(Optional) --
Step 6	**end** **Example:** `Router(config)# end`	(Required) Exits the current configuration mode and returns to privileged EXEC mode.

Configuring NetFlow Export from the Main NetFlow Cache using the Version 9 Export Format

SUMMARY STEPS

1. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfEIExportVersion**.*type* **unsigned** *version* **cnfEIPeerAS.** *type* **integer** *truth-value* **cnfEIBgpNextHop**.*type* **integer** *truth-value*
2. **snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfEICollectorStatus.** *type* **.** *address-type* **.** *ip-version* **.** *ip-address* **.** *port* **integer** [**4** | **6**]
3. Repeat Step 2 to add another collector

DETAILED STEPS

	Command or Action	Purpose
Step 1	**snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfEIExportVersion**.*type* **unsigned** *version* **cnfEIPeerAS.** *type* **integer** *truth-value* **cnfEIBgpNextHop**.*type* **integer** *truth-value* **Example**: `workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfEIExportVersion.0 unsigned 9 cnfEIPeerAS.0 integer 1 cnfEIBgpNextHop.0 integer 1`	(Required) Specifies the export format and that the export statistics include peer autonomous system and BGP-related information. • The values for the *type* argument are: ◦ Main--0 ◦ AS--1 ◦ Protocol Port--2 ◦ Source Prefix--3 ◦ Destination Prefix--4 ◦ prefix--5 ◦ Destination Only--6 ◦ Source Destination--7 ◦ Full Flow--8 ◦ AS ToS--9 ◦ Protocol Port ToS--10 ◦ Source Prefix ToS--11 ◦ Destination Prefix Tos--12 ◦ Prefix Tos--13 ◦ Prefix Port--14 ◦ BGP Nexthop Tos--15 • The values for the *version* argument are: ◦ 5--Version 5 export format. The number of records stored in the datagram is a variable between 1 and 30 for the Version 5 export format. ◦ 9--Version 9 export format. • The values for the *truth-value* argument are: ◦ 1--enable the keyword ◦ 2--disable the keyword

	Command or Action	Purpose
Step 2	**snmpset -c private -m all -v2c** [*ip-address* | *hostname*] **cnfEICollectorStatus.** *type* **.** *address-type* **.** *ip-version* **.** *ip-address* **.** *port* **integer** [**4** | **6**] **Example:** `workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfEICollectorStatus. 0.1.4.10.0.19.2.3 integer 4`	(Required) Enables the exporting of information in NetFlow cache entries. • Values the *type* argument are: ◦ Main--0 ◦ AS--1 ◦ Protocol Port--2 ◦ Source Prefix--3 ◦ Destination Prefix--4 ◦ prefix--5 ◦ Destination Only--6 ◦ Source Destination--7 ◦ Full Flow--8 ◦ AS ToS--9 ◦ Protocol Port ToS--10 ◦ Source Prefix ToS--11 ◦ Destination Prefix Tos--12 ◦ Prefix Tos--13 ◦ Prefix Port--14 ◦ BGP Nexthop Tos--15 • The *address-type*, and *ip-version* arguments specify the type of IP address. ◦ The *address-type* argument is 1. ◦ The *ip-version* argument is the length in bytes of the address. Currently IPv4 is the only type that is supported, so the ip-version value should be 4 (four bytes in an IPv4 IP address). • The *ip-address* variable specifies the IPv4 IP address of the collector. • The *port* argument is the UDP port the collector is listening on for NetFlow data. • The [**4** | **6**] keywords create and remove the NetFlow collector. ◦ The 4 keyword creates the collector in the router's configuration, and activates the collector. ◦ The 6 keyword removes the collector from router's configuration.
Step 3	Repeat Step 2 to add another collector	(Optional) --

Configuration Examples using SNMP and the NetFlow MIB to Monitor NetFlow Data

Configuring the Minimum Mask for a Source Prefix Aggregation Scheme using SNMP Example

The following example enables a **Source-Prefix** aggregation cache and sets the source prefix mask to 16 bits.

```
workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCICacheEnable.3 integer 1
CISCO-NETFLOW-MIB::cnfCICacheEnable.sourcePrefix = INTEGER: true(1)
workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCIMinSourceMask.3 unsigned 16
CISCO-NETFLOW-MIB::cnfCIMinSourceMask.sourcePrefix = Gauge32: 16
```

Configuring NetFlow Data Export for the Source Prefix Aggregation Scheme using SNMP Example

The following example enables a **Source-Prefix** aggregation cache and configures NetFlow Data Export for the aggregation cache.

```
workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCICacheEnable.3 integer 1
CISCO-NETFLOW-MIB::cnfCICacheEnable.sourcePrefix = INTEGER: true(1)
workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfEICollectorStatus.
3.1.4.10.0.19.2.3 integer 4
CISCO-NETFLOW-MIB::cnfEICollectorStatus.sourcePrefix.ipv4.".....".3 = INTEGER:
createAndGo(4)
```

Configuring a NetFlow Minimum Mask for a Prefix Aggregation Cache using SNMP Example

The following example enables a **Prefix** aggregation cache and sets the prefix mask to 16 bits.

```
workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCICacheEnable.5 integer 1
CISCO-NETFLOW-MIB::cnfCICacheEnable.prefix = INTEGER: true(1)
workstation% snmpset -c private -m all -v2c 10.4.9.14 cnfCIMinSourceMask.5
 unsigned 16
CISCO-NETFLOW-MIB::cnfCIMinSourceMask.prefix = Gauge32: 16
```

Using SNMP to Gather Flow Information From the Router Example

The following examples show how to retrieve NetFlow status and statistics using SNMP.

Retrieving Netflow Statistics using SNMP

This command will retrieve the Netflow Statistics from the main cache using the MIB.

```
workstation% snmpget -c public -v2c 10.4.9.14 cnfPSPacketSizeDistribution.0
cnfPSPacketSizeDistribution.0 =
```

```
00 00  00 00   03 e8  00 00    00 00  00 00   00 00  00 00
00 00  00 00   00 00  00 00    00 00  00 00   00 00  00 00
00 00  00 00   00 00  00 00    00 00  00 00   00 00  00 00
00 00  00 00
```

The IP packet size distribution values are in the same order as shown in the CLI, with each pair of bytes representing a value of 1000 times the respective value in the CLI.

For example, for the packet range 65-96, the byte pair is 0x03e8 which is 1000 times 1. So to obtain the same values as the CLI, divide the value by 1000.

View the NetFlow Main Cache Timeout Values using SNMP

This command will retrieve the cache timeout values from the main cache using the MIB.

```
workstation% snmpget -c public -v2c 10.4.9.14 cnfCIActiveFlows.0 cnfCIInactiveFlows.0
cnfCIActiveTimeOut.0 cnfCIInactiveTimeOut.0
CISCO-NETFLOW-MIB::cnfCIActiveFlows.main = Gauge32: 1
CISCO-NETFLOW-MIB::cnfCIInactiveFlows.main = Gauge32: 3999
CISCO-NETFLOW-MIB::cnfCIActiveTimeOut.main = Gauge32: 60 minutes
CISCO-NETFLOW-MIB::cnfCIInactiveTimeOut.main = Gauge32: 30 seconds
```

Additional References

Related Documents

Related Topic	Document Title
Overview of Cisco IOS NetFlow	"Cisco IOS NetFlow Overview"
The minimum information about and tasks required for configuring NetFlow and NetFlow Data Export	"Getting Started with Configuring NetFlow and NetFlow Data Export"
Tasks for configuring NetFlow to capture and export network traffic data	"Configuring NetFlow and NetFlow Data Export"
Tasks for configuring Configuring MPLS Aware NetFlow	Configuring MPLS Aware NetFlow
Tasks for configuring MPLS egress NetFlow accounting	Configuring MPLS Egress NetFlow Accounting and Analysis
Tasks for configuring NetFlow input filters	"Using NetFlow Filtering or Sampling to Select the Network Traffic to Track"
Tasks for configuring Random Sampled NetFlow	"Using NetFlow Filtering or Sampling to Select the Network Traffic to Track"
Tasks for configuring NetFlow aggregation caches	"Configuring NetFlow Aggregation Caches"
Tasks for configuring NetFlow BGP next hop support	"Configuring NetFlow BGP Next Hop Support for Accounting and Analysis"
Tasks for configuring NetFlow multicast support	"Configuring NetFlow Multicast Accounting"

Related Topic	Document Title
Tasks for detecting and analyzing network threats with NetFlow	Detecting and Analyzing Network Threats With NetFlow
Tasks for configuring NetFlow Reliable Export With SCTP	NetFlow Reliable Export With SCTP
Tasks for configuring NetFlow Layer 2 and Security Monitoring Exports	"NetFlow Layer 2 and Security Monitoring Exports"
Tasks for configuring the NetFlow MIB and Top Talkers feature	"Configuring NetFlow Top Talkers using Cisco IOS CLI Commands or SNMP Commands"
Information for installing, starting, and configuring the CNS NetFlow Collection Engine	"Cisco CNS NetFlow Collection Engine Documentation"

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

MIBs	MIBs Link
• CISCO-NETFLOW-MIB.my	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL (requires CCO login account): http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	--

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Feature Information for Configuring SNMP and using the NetFlow MIB to Monitor NetFlow Data

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 1 Feature Information for Configuring the NetFlow Top Talkers Feature using the Cisco IOS CLI or SNMP Commands

Feature Name	Releases	Feature Configuration Information
NetFlow MIB	12.3(7)T, 12.2(25)S 12.2(27)SBC 12.2(33)SRD	The NetFlow MIB feature provides MIB objects to allow users to monitor NetFlow cache information, the current NetFlow configuration, and statistics. The following command was introduced by this feature: **ip flow-cache timeout**.

Glossary

AS --autonomous system. A collection of networks under a common administration sharing a common routing strategy. Autonomous systems are subdivided by areas. An autonomous system must be assigned a unique 16-bit number by the Internet Assigned Numbers Authority (IANA).

BGP --Border Gateway Protocol. Interdomain routing protocol that replaces Exterior Gateway Protocol (EGP). A BGP system exchanges reachability information with other BGP systems. BGP is defined by RFC 1163.

BGP next hop --IP address of the next hop to be used to reach a specific destination.

CEF --Cisco Express Forwarding. A Layer 3 IP switching technology that optimizes network performance and scalability for networks with large and dynamic traffic patterns.

dCEF --distributed Cisco Express Forwarding. A type of CEF switching in which line cards (such as Versatile Interface Processor (VIP) line cards) maintain identical copies of the forwarding information base (FIB) and adjacency tables. The line cards perform the express forwarding between port adapters; this relieves the Route Switch Processor of involvement in the switching operation.

MIB --Management Information Base. Database of network management information that is used and maintained by a network management protocol, such as Simple Network Management Protocol (SNMP) or the Common Management Information Protocol (CMIP). The value of a MIB object can be changed or retrieved using SNMP or CMIP commands, usually through a GUI network management system. MIB objects are organized in a tree structure that includes public (standard) and private (proprietary) branches.

NetFlow --A Cisco IOS application that provides statistics on packets flowing through the router. It is emerging as a primary network accounting and security technology.

NetFlow Aggregation --A NetFlow feature that lets you summarize NetFlow export data on an IOS router before the data is exported to a NetFlow data collection system such as the NetFlow Collection Engine. This feature lowers bandwidth requirements for NetFlow export data and reduces platform requirements for NetFlow data collection devices.

NetFlow Collection Engine (formerly NetFlow FlowCollector)--A Cisco application that is used with NetFlow on Cisco routers and Catalyst series switches. The NetFlow Collection Engine collects packets from the router that is running NetFlow and decodes, aggregates, and stores them. You can generate reports on various aggregations that can be set up on the NetFlow Collection Engine.

NetFlow v9 --NetFlow export format Version 9. A flexible and extensible means for carrying NetFlow records from a network node to a collector. NetFlow Version 9 has definable record types and is self-describing for easier NetFlow Collection Engine configuration.

NMS --network management system. A system responsible for managing at least part of a network. An NMS is generally a reasonably powerful and well-equipped computer, such as an engineering workstation. NMSs communicate with agents to help keep track of network statistics and resources.

SNMP --Simple Network Management Protocol. A network management protocol used almost exclusively in TCP/IP networks. SNMP provides a means to monitor and control network devices, and to manage configurations, statistics collection, performance, and security.

SNMP communities --An authentication scheme that enables an intelligent network device to validate SNMP requests.

ToS byte --type of service byte. Second byte in the IP header that indicates the desired quality of service for a particular datagram.