

Netflow Configuration Guide for Cisco NCS 5500 Series Routers, IOS XR Release 26.1.x, 26.2.x, 26.3.x

First Published: 2026-03-03

Last Modified: 2026-09-11

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

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

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

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

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

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

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

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

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

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

© 2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface vii

Changes to This Document vii

Communications, Services, and Additional Information vii

CHAPTER 1

New and Changed Feature Information 1

New and Changed Information 1

CHAPTER 2

NetFlow Overview 3

Recording of Packet Flows in NetFlow 3

Prerequisites for Configuring NetFlow 5

Restrictions for Configuring NetFlow 5

Information About Configuring NetFlow 7

NetFlow Overview 7

Exporter Map Overview 8

Monitor Map Overview 9

Sampler Map Overview 10

How to Configure NetFlow on Cisco IOS XR Software 11

Configuring an Exporter Map 11

Configuring a Sampler Map 14

Configuring a Monitor Map 15

Applying a Monitor Map and a Sampler Map to a Physical Interface 18

Applying a Monitor Map and a Sampler Map to a Layer 2 Bundle Interface 19

Clearing NetFlow Data 20

Configure NetFlow Collection of MPLS Packets with IPv6 Fields 20

Drop Codes on NetFlow 24

Additional References 24

CHAPTER 3 **NetFlow Supported Features** 27

In-line Modification of Netflow Configuration 27

Use Case 28

Options Template Overview 29

Flow Filter 31

Restrictions 31

Configuring Flow Filter 31

IPFIX 34

Collect Additional BGP Information Elements for MPLS IPv4 and IPV6 Using IPFIX 35

Configuring IPFIX 38

IPFIX Enablement for SRv6 and Services over SRv6 Core 41

BGP community and AS path information elements for IPFIX 46

Guidelines for exporting BGP and AS attributes using IPFIX 47

Configure BGP community and AS path export for IPFIX 47

IP Flow Information Export (IPFIX) 315 48

IPFIX 315 Implementation Considerations 49

Configuring IPFIX 315 50

NetFlow Configuration Submodes 52

Flow Monitor Map Configuration Submode 52

Flow Exporter Map Configuration Submode 53

Flow Exporter Map Version Configuration Submode 54

Sampler Map Configuration Submode 55

Enabling the NetFlow BGP Data Export Function 55

MPLS Flow Monitor with IPv4 and IPv6 Support 55

MPLS Cache Reorganization to Support Both IPv4 and IPv6 55

MPLS Packets with IPv6 Flows 56

Monitor GTP-U Traffic in 5G Network 58

Netflow Full Packet Capture 74

Configuring Netflow Full Packet Capture 74

CHAPTER 4 **IPFIX** 75

Collect Additional BGP Information Elements for MPLS IPv4 and IPV6 Using IPFIX 76

Configuring IPFIX 79

IPFIX Enablement for SRv6 and Services over SRv6 Core 82
IP Flow Information Export (IPFIX) 315 87
IPFIX 315 Implementation Considerations 88
Configuring IPFIX 315 89

CHAPTER 5 **Configuring sFlow** 93
sFlow Agent 93
Guidelines and Limitations for sFlow 95
Default Settings for sFlow 96
Configuring sFlow 96
Configuring Exporter Map 97
Configuring Monitor Map 97
Configuring Sampler Map 98
Configuring sFlow on an Interface 99
Enabling sFlow on a Line Card 99
Verify sFlow Configuration 99
Discard Codes on sFlow 100

CHAPTER 6 **Scenario A: Traffic Monitoring Without NetFlow and sFlow** 103
Scenario B: Traffic Monitoring With NetFlow and sFlow 103



Preface

This preface contains these sections:

- Changes to This Document, on page vii
- Communications, Services, and Additional Information, on page vii

Changes to This Document

This table lists the technical changes made to this document since it was first released.

Table 1: Changes to This Document

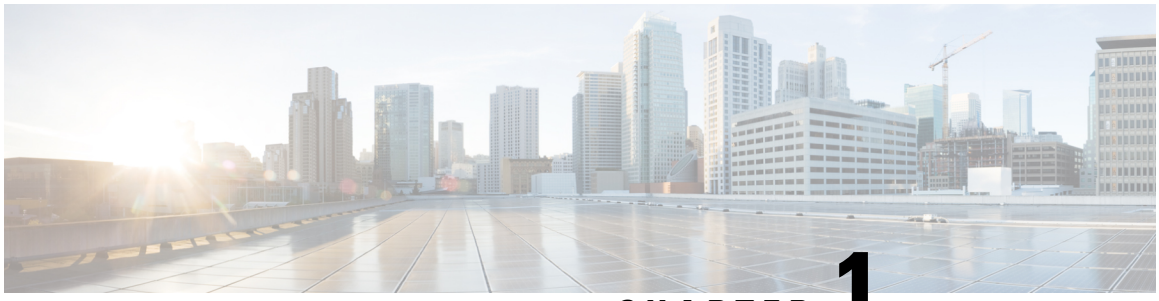
Date	Summary
September 2026	Republished for 26.3.1
June 2026	Republished for 26.2.1
February 2026	Initial release of this document

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business results you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



CHAPTER 1

New and Changed Feature Information

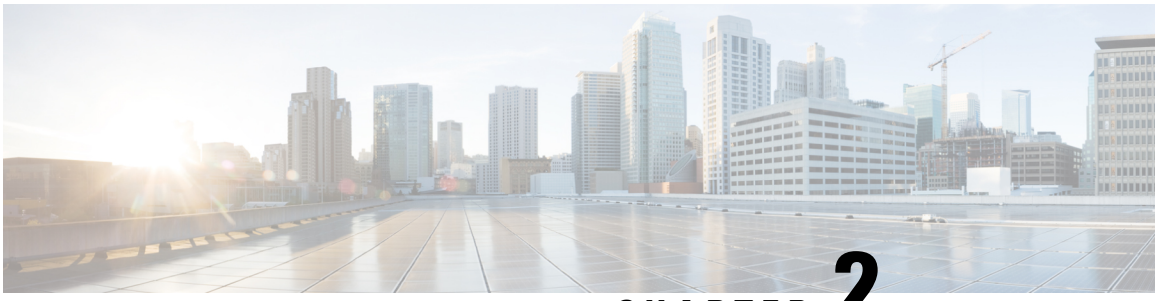
This table summarizes the new and changed feature information for the *Netflow Configuration Guide for Cisco NCS 5500 Series Routers*, and tells you where they are documented.

• New and Changed Information, on page 1

New and Changed Information

Table 2: New and Changed Features

Feature	Description	Changed in Release	Where Documented



CHAPTER 2

NetFlow Overview

A NetFlow flow is a unidirectional sequence of packets that arrive on a single interface, and have the same values for key fields.

NetFlow is useful for the following:

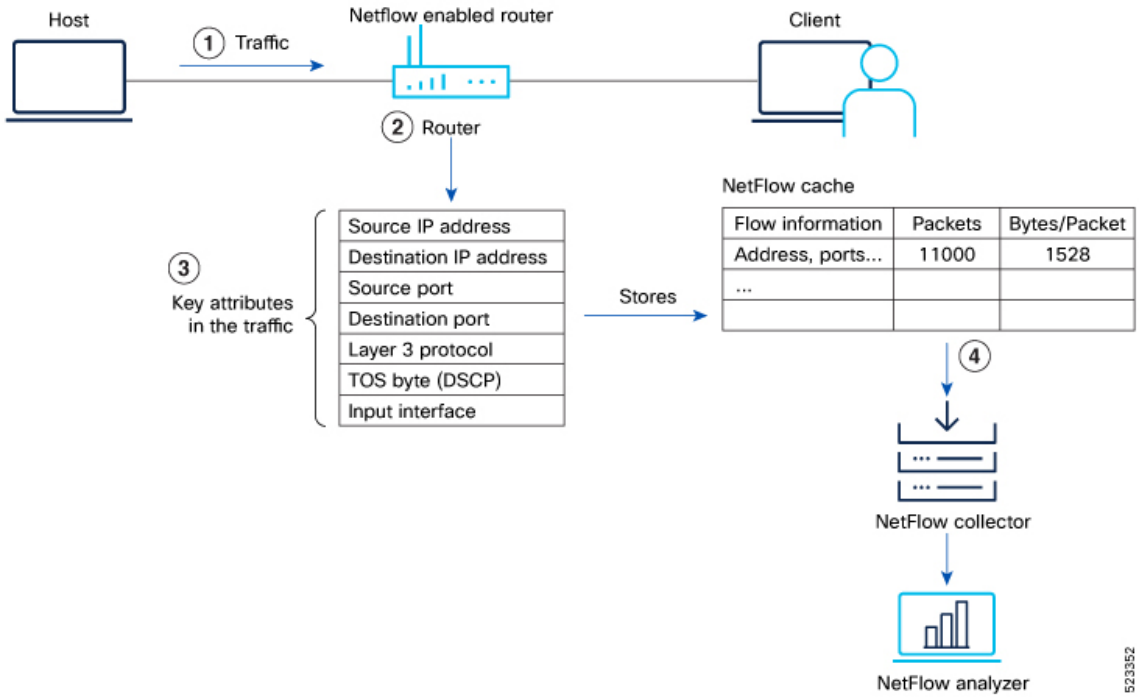
- Accounting/Billing—NetFlow data provides fine grained metering for highly flexible and detailed resource utilization accounting.
- Network Planning and Analysis—NetFlow data provides key information for strategic network planning.
- Network Monitoring—NetFlow data enables near real-time network monitoring capabilities.

- Recording of Packet Flows in NetFlow, on page 3
- Prerequisites for Configuring NetFlow, on page 5
- Restrictions for Configuring NetFlow, on page 5
- Information About Configuring NetFlow, on page 7
- How to Configure NetFlow on Cisco IOS XR Software, on page 11
- Drop Codes on NetFlow, on page 24
- Additional References, on page 24

Recording of Packet Flows in NetFlow

The packet in NetFlow is recorded as follows:

Figure 1: Packet Flows in NetFlow



1 Flow Creation
2 Datagram Generation
3 Data Export
4 Analysis and Reporting

In NetFlow, the focus is on recording and collecting full packet flows in the network traffic data. When NetFlow is configured on the router, the router collects flow data by extracting key field attributes from the packet streams, and generates a flow record. This record, along with accounting information, is stored in the database or NetFlow Cache. The extracted records, once sampled, are exported to one or more NetFlow collectors via the UDP transport layer protocol. This exported data has several purpose: enterprise accounting and ISP billing, and so on.

Here's how NetFlow handles the recording of packet flows:

1. Flow Creation: NetFlow creates flow records by monitoring network traffic passing through the router. As a packet stream traverses a router interface, the packets are collected and an internal header is appended. These packets are dispatched to the line card's CPU, which generate a flow record. The router extracts pertinent header details from the packets and creates cache entries. The packets are subject to a policer, which helps protect the internal control plane. With each subsequent arrival of a packet from the same flow, the cache entry is updated. Flow records persist within the line card's cache until they age out due to timer expiration.

   When the expiry of the set timer occurs, the NetFlow is generated. There are timers (two of them) running for flow aging.

   - The active timer signifies the maximum allowable duration for a particular cache entry's existence, even if matched by received sampled packets.

- The inactive timer represents the duration without receipt of a sampled packet corresponding to a specific cache entry.

2. Datagram Generation: The NetFlow agent generates NetFlow datagrams that contain information about the packets. These datagrams include details such as source and destination IP addresses, port numbers, protocol information, and various flow statistics.
3. Data Export: The NetFlow datagrams are periodically exported from the NetFlow agent to a designated NetFlow collector or analyzer. The export can be done using protocols like UDP or TCP, and the datagrams are typically sent in a structured format like IPFIX or JSON.

   A flow record is sent to the NetFlow collector in the following scenarios:

   - The flow has been inactive or active for an extended period.
   - The user triggers the export of the flow.
   - The flow concludes, which is particularly relevant when TCP connections are terminated.

4. Analysis and Reporting: Upon receiving the NetFlow data, the NetFlow collector or analyzer processes and analyzes the information. It aggregates the sampled data to provide statistical insights into network traffic, including top talkers, protocol distribution, traffic patterns, and other metrics.

Prerequisites for Configuring NetFlow

To perform these configuration tasks, your Cisco IOS XR software system administrator must assign you to a user group associated with a task group that includes the corresponding command task IDs. If you need assistance with your task group assignment, contact your system administrator.

Restrictions for Configuring NetFlow

Consider these restrictions when configuring NetFlow in Cisco IOS XR software:

Tip Do not use the management interface to export the NetFlow packets.

- NetFlow can be configured only in the ingress direction.
- Netflow v9, IPFIX, and IPFIX 315 support a maximum of two sampler maps.
- A source interface must always be configured. If you do not configure a source interface, the exporter will remain in a disabled state.
- Only export format Version 9 and IPFIX is supported.
- A valid record map name must always be configured for every flow monitor map.
- NetFlow is not supported on Bridge Virtual Interface (BVI).
- NetFlow on sub-interface routed via BVI is not supported.

- Destination-based Netflow accounting is not supported, only IPv4, IPv6 and MPLS record types are supported under monitor-map.
- Output interface field is not updated in data and flow records when the traffic is routed through ACL based forwarding (ABF).
- Output interface, source, and destination prefix lengths fields is not updated in data and flow records for multicast traffic.
- Output interface, source and destination prefix lengths fields are not set in data and flow records for GRE transit traffic.
- In-line modification of flow attribute record of NetFlow configuration is not supported.
- For Netflow IPFIX315, configure the **hw-module profile netflow ipfix315** command.
- If IPFIX315 is enabled on a line card then all the ports on that line card should have IPFIX315 configured.
- For **hw-module profile qos hqos-enable**, NetFlow does not give the output interface for cases like L2 bridging, xconnect, IPFIX, and so on.
- L4 header port numbers are supported only for TCP and UDP.
- NetFlow does not give the output interface for traffic terminating on GRE tunnel.
- If full packet capture is disabled, then NetFlow captures only IPv4 and IPv6 packets. To enable packet flow for IPv4, IPv6, and L2VPN psuedo wire packets, enable the **hw-module profile netflow fpc-enable location** command and perform a reload.
- When SRv6 accounting (accounting prefixes in IPv6 mode **per-prefix per-nexthop SRv6-locators**) is enabled, the output interface field is not updated in data and flow records.
- From Release 24.2.1, Cisco NCS 5700 series line cards and fixed routers do not support NetFlow export on L2 transport or EVPN bridge domain interfaces. When you enable NetFlow export on L2 transport, the IOS XR software either does not display any output or displays incomplete IPv4 sampling. This limitation affects only IPv4 and ARP packets.

Scale Restrictions

Maximum Sampler Rate

- For NC57 line card, a maximum sampler rate of 1:2000 can be supported.
- A rate of 1:4000 is recommended for other line cards if NetFlow needs to be configured on all interfaces.

Maximum Monitor Maps per Interface

- An interface can be configured with a maximum of 3 monitor maps at a time:
  - Record ipv4
  - Record ipv6
  - Record mpls
- For IPFIX-315, only the record datalinksectiondump can be associated with the interface.
- For sFlow, only the record sflow can be associated with the interface.

Flow Cache- Maximum supported flow cache is 1000000.

Information About Configuring NetFlow

NetFlow Overview

Netflow is used to create a statistical view of the flow matrix from the router - at the beginning of Netflow Overview section before explanation of flows.

A flow is exported as part of a NetFlow export User Datagram Protocol (UDP) datagram under these circumstances:

- The flow has been inactive or active for too long.
- The flow cache is getting full.
- One of the counters (packets and or bytes) has wrapped.
- The user forces the flow to export.

NetFlow export UDP datagrams are sent to an external flow collector device that provides NetFlow export data filtering and aggregation. The export of data consists of expired flows and control information.

The NetFlow infrastructure is based on the configuration and use of these maps:

- Exporter map
- Monitor map
- Sampler map

From Release 25.2.1, you can enable NetFlow export on L2 transport or EVPN bridge domain interfaces on Cisco NCS 5700 series line cards and fixed routers.

Cross AFI BGP NH information elements

Cross AFI BGP NH information elements specifies the next hop IP address for different network layer protocols in BGP routing. These elements ensure

- proper routing across diverse network environments by indicating the appropriate next hop based on the Address Family Identifier (AFI) and
- its Subsequent Address Family Identifier (SAFI).

Table 3: Feature History Table

Feature Name	Release Information	Feature Description
Cross AFI BGP NH Information Element	Release 25.1.1	Introduced in this release on: NCS 5500 fixed port routers; NCS 5700 fixed port routers; NCS 5500 modular routers (NCS 5500 line cards; NCS 5700 line cards [Mode: Compatibility; Native]) IPv4 or IPv6 flows in BGP can now handle next-hop Information Element (IE) across different address families, such as IPv4 and IPv6. This is particularly useful in scenarios where IPv4 and IPv6 networks need to interoperate. These IEs are added to the existing NetFlow or IPFIX template for `record ipv4` and all the IPv4 variant record types: • BgpNextHopIPv6Address (IE 63) • IpNextHopIPv6Address (IE 62) • IpNextHopIPv4Address (IE 15) These IEs are added to the existing NetFlow or IPFIX template for `record ipv6` and all the IPv6 variant record types: • BgpNextHopIPv4Address (IE 18) • IpNextHopIPv6Address (IE 62) • IpNextHopIPv4Address (IE 15) These IEs provide a detailed and structured data that is essential for various network operations and analyses. The feature uses the exiting CLI commands. For more information see, IPFIX Enablement for SRv6 and Services over SRv6 Core.

Exporter Map Overview

An exporter map contains user network specification and transport layer details for the NetFlow export packet. The **flow exporter-map** command allows you to configure collector and version attributes. You can configure these collector information:

- Export destination IP address
- DSCP value for export packet
- Source interface
- UDP port number (This is where the collector is listening for NetFlow packets.)

- Transport protocol for export packets

Note In Cisco IOS XR Software, UDP is the only supported transport protocol for export packets.

Note NetFlow export packets use the IP address that is assigned to the source interface. If the source interface does not have an IP address assigned to it, the exporter will be inactive.

You can also configure these export version attributes:

- Template timeout
- Template data timeout
- Template options timeout
- Interface table timeout
- Sampler table timeout

Note A single flow monitor map can support up to eight exporters.

Monitor Map Overview

A monitor map contains name references to the flow record map and flow exporter map. Monitor maps are applied to an interface. You can configure these monitor map attributes:

- Number of entries in the flow cache
- Type of cache (permanent or normal). Permanent caches do not have their entries removed from the cache unless they are explicitly cleared by the user
- Active flow timeout
- Inactive flow timeout
- Update timeout
- Default timeouts
- Record type of packets sampled and collected

Note The record name specifies the type of packets that NetFlow samples as they pass through the router. Currently, MPLS, IPv4, and IPv6 packet sampling is supported.

Note The active flow and inactive flow timeouts are associated with a normal cache type. The update timeout is associated with the permanent cache type.

Sampler Map Overview

Table 4: Feature History Table

Feature Name	Release Information	Description
Enhanced NetFlow Sampling Rate of 1:2048 (2K)	Release 7.4.1	You can configure a sampling rate of 1:2048 on NC57 line card when the line card is configured in the native mode. Previously, the line card supported configuring Netflow sampling rate of 1:4096(4K), 1:8192(8K), and 1:16384(16K) The command **random 1 out-of** is modified to support the new sampling rate.

The sampler map specifies the rate at which packets (one out of n packets) are sampled. The sampler map configuration is typically geared for high-speed interfaces to optimize CPU utilization. To achieve this, start by setting the sampling rate after evaluating your network parameters such as traffic rate, number of total flows, cache size, active and inactive timers.

- The maximum supported sampling rate is 1:1, where every packet is processed.
- The minimum supported sampling rate is 1:65,536, indicating that only one out of every 65,536 packets is processed.

Consider these points before applying sampler map:

Note While caching netflow traffic over bundle interface, a deviation in flow monitor cache entries is observed.

The deviation is not always consistent, and the acceptable limit is up to 15%

Consider these points before applying sampler map:

- You must remove the existing netflow configuration before applying a new sampler map on an already existing netflow interface configuration.
- Sub-interfaces and physical interfaces under a port must have the same sampler map configuration.

How to Configure NetFlow on Cisco IOS XR Software

The steps that follow provide a general overview of NetFlow configuration:

Note We recommend that you not use the default ethernet VLAN (VLAN-1) in any of your network configurations. Traffic tagged with VLAN-1 may cause conflicts with other configurations.

Procedure

Step 1 Create and configure an exporter map.

Step 2 Create and configure a monitor map and a sampler map.

Note
The monitor map must reference the exporter map you created in Step 1. If you do not apply an exporter-map to the monitor-map, the flow records are not exported, and aging is done according to the cache parameters specified in the monitor-map.

Step 3 Apply the monitor map and sampler map to an interface.

These steps are described in detail in these sections:

Configuring an Exporter Map

Configure an exporter map and apply it to the monitor map with the **flow monitor-map** *map_name* **exporter** *map_name* command. You can configure the exporter map prior to configuring the monitor map, or you can configure the monitor map first and then configure and apply an exporter map later on.

Note Cisco IOS XR Software supports the configuration of a single collector only in the exporter map.

The steps that follow describe how to create and configure an exporter map and enable exporting of the sampler table or the interface table.

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router#configure
```

Enters global configuration mode.

Step 2 **flow exporter-map** *map_name*

Example:

```
RP/0/RP0/CPU0:router(config)#flow exporter-map expmap-dtxr2
```

Creates an exporter map, configures the exporter map name, and enters flow exporter map configuration mode.

Step 3 **destination** *hostname_or_IP_address* [**vrf** *vrf-name*]

Example:

```
RP/0/RP0/CPU0:router(config-fem)# destination 1.76.31.1
```

Configures the export destination for the flow exporter map. The destination can be a hostname, a VRF, or an IPv4/IPv6 address.

Step 4 **dscp** *dscp_value*

Example:

```
RP/0/RP0/CPU0:router(config-fem)# dscp 10
```

(Optional) Specifies the differentiated services codepoint (DSCP) value for export packets. Replace the *dscp_value* argument with a value in the range from 0 through 63.

Step 5 **source** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-fem)# source Loopback 0
```

Specifies a source interface, in the format *type interface-path-id*.

Step 6 **transport udp** *port*

Example:

```
RP/0/RP0/CPU0:router(config-fem)# transport udp 5999
```

(Optional) Specifies the destination port for UDP packets. Replace *port* with the destination UDP port value, in the range from 1024 through 65535.

Step 7 **version v9**

Example:

```
RP/0/RP0/CPU0:router(config-fem-ver)# version v9
```

(Optional) Enters flow exporter map version configuration submode.

Step 8 **options** {**interface-table** | **sampler-table** | **vrf-table**} [**timeout** *seconds*]

Example:

```
RP/0/RP0/CPU0:router(config-fem-ver)# options sampler-table timeout 1800
```

(Optional) Configures the export timeout value for the sampler table. Replace *seconds* with the export timeout value, in the range from 1 through 604800 seconds.

Default is 1800 seconds.

Step 9 **template** [**data** | **options**] **timeout** *seconds*

Example:

```
RP/0/RP0/CPU0:router(config-fem-ver)# template data timeout 600
```

(Optional) Configures the export period for data packets. Replace *seconds* with the export timeout value, in the range from 1 through 604800 seconds.

Step 10 **commit**

Step 11 **exit**

Example:

```
RP/0/RP0/CPU0:router(config-fem-ver)# exit
```

Exits flow exporter map version configuration submode.

Step 12 **exit**

Example:

```
RP/0/RP0/CPU0:router(config)# exit
```

Enters XR EXEC mode.

Step 13 **show flow exporter-map** *map_name*

Example:

```
RP/0/RP0/CPU0:router# show flow exporter-map expmap-dtxr2

Flow Exporter Map : expmap-dtxr2
-------------------------------------------------
Id                  : 1
DestinationIpAddr   : 1.76.31.1
VRFName             : default
SourceIfName        : Loopback0
SourceIpAddr        : 10.200.58.1
DSCP                : 10
TransportProtocol   : UDP
TransportDestPort   : 5999

Export Version: 9
   Common Template Timeout : 1800 seconds
   Options Template Timeout : 1800 seconds
   Data Template Timeout : 600 seconds
   Interface-Table Export Timeout : 1800 seconds
   Sampler-Table Export Timeout : 0 seconds
   VRF-Table Export Timeout : 0 seconds
```

Displays exporter map data.

Example

This example shows how to create a new flow exporter map called “fem1,” which uses the version 9 (V9) export format for NetFlow export packets. The data template flow-set is inserted into the V9 export packets once every 10 minutes, and the options interface table flow-set is inserted into the V9 export packet. The export packets are sent to the flow collector destination 10.1.1.1, where the source address is identical to the interface IP address of Loopback 0. The UDP destination port is 1024, and the DSCP value is 10:

```
RP/0/RP0/CPU0:router(config)# flow exporter-map fem1
```

```
RP/0/RP0/CPU0:router(config-fem)# destination 10.1.1.1
RP/0/RP0/CPU0:router(config-fem)# source Loopback 0
RP/0/RP0/CPU0:router(config-fem)# transport udp 1024
RP/0/RP0/CPU0:router(config-fem)# dscp 10
RP/0/RP0/CPU0:router(config-fem)# exit
RP/0/RP0/CPU0:router(config-fem)# version v9
RP/0/RP0/CPU0:router(config-fem-ver)# template data timeout 600
RP/0/RP0/CPU0:router(config-fem-ver)# options interface-table
RP/0/RP0/CPU0:router(config-fem-ver)# exit
```

Configuring a Sampler Map

Procedure

Step 1 **configure**

Example:

```
RP/0/RSP0/CPU0:router#configure
```

Enters global configuration mode.

Step 2 **sampler-map** *map_name*

Example:

```
RP/0/RP0/CPU0:router(config)# sampler-map onein8k
RP/0/RP0/CPU0:router(config-sm)#
```

Creates a sampler map and enters sampler map configuration mode.

Step 3 **random 1 out-of** *sampling_interval*

Example:

```
RP/0/RP0/CPU0:router(config-sm)# random 1 out-of 8000
```

Configures the sampling interval to use random mode for sampling packets. Replace the *sampling_interval* argument with a number, in the range from 1 through 65535 units.

Note
The sampling interval of 1:1000 packets is supported.

Step 4 **commit**

Step 5 **exit**

Example:

```
RP/0/RP0/CPU0:router(config-sm)# exit
```

Exits sampler map configuration mode and enters the XR Config mode.

Step 6 **exit**

Example:

```
RP/0/RP0/CPU0:router(config)# exit
```

Exits the mode and enters XR EXEC mode.

Step 7 **show sampler-map** *map_name*

Example:

```
RP/0/RP0/CPU0:router#show sampler-map onein8k

Sampler Map : onein8k
----------------------------------------------
Id:      1
Mode:    Random (1 out of 8000 Pkts)
```

Displays sampler map data.

Example

This example shows how to create a new sampler map called "fsm1," which samples 1 out of 65535 packets:

```
RP/0/RP0/CPU0:router# sampler-map fsm1
RP/0/RP0/CPU0:router(config-sm)# random 1 out-of 65535
RP/0/RP0/CPU0:router(config)# exit
```

Configuring a Monitor Map

Procedure

Step 1 **configure**

Example:

```
RP/0/RSP0/CPU0:router#configure
```

Enters global configuration mode.

Step 2 **flow monitor-map** *map_name*

Example:

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm-ipv4-dtxr2
RP/0/RP0/CPU0:router(config-fmm)#
```

Creates a monitor map and configures a monitor map name and enters flow monitor map configuration submode.

Step 3 Do one of the following:

- **record ipv4**
- **record ipv4** [**peer as**]
- **record ipv6**
- **record mpls** [**labels** *number*]
- **record mpls** [**ipv4-fields**] [**labels** *number*]
- **record mpls** [**ipv6-fields**] [**labels** *number*]

- **record mpls** [**ipv4-ipv6-fields**] [**labels** *number*]

Example:

```
RP/0/RP0/CPU0:router(config-fmm)# record ipv4
```

Configures the flow record map name for IPv4, IPv6, or MPLS.

- Use the **record ipv4** command to configure the flow record map name for IPv4. By default, you collect and export the originating autonomous system (AS) numbers.
- Use the **record ipv4** [**peer-as**] command to record peer AS. Here, you collect and export the peer AS numbers.

Note
Ensure that the **bgp attribute-download** command is configured. Else, no AS is collected when the **record ipv4** or **record ipv4 peer-as** command is configured.

- Use the **record ipv6** command to configure the flow record map name for IPv6.
- Use the **record mpls labels** command with the *number* argument to specify the number of labels that you want to aggregate. By default, MPLS-aware NetFlow aggregates the top six labels of the MPLS label stack. The maximum value is 6.
- Use the **record mpls ipv4-fields** command to collect IPv4 fields in the MPLS-aware NetFlow.
- Use the **record mpls ipv6-fields** command to collect IPV6 fields in the MPLS-aware NetFlow.
- Use the **record mpls ipv4-ipv6-fields** command to collect IPv4 and IPv6 fields in the MPLS-aware NetFlow.

  Note
  For the **outbundlemember** option to be effective; you must configure monitor-map as following:

  ```
  flow monitor-map nfmpls
  record mpls ipv4-ipv6-fields
  option outbundlemember
  ```

Step 4 **cache entries** *number*

Example:

```
RP/0/RP0/CPU0:router(config-fmm)# cache entries 65535
```

(Optional) Configures the number of entries in the flow cache. Replace the *number* argument with the number of flow entries allowed in the flow cache, in the range from 4096 through 1000000.

The default number of cache entries is 65535.

Step 5 **cache permanent**

Example:

```
RP/0/RP0/CPU0:router(config-fmm)# flow monitor-map fmm cache permanent
```

(Optional) Disables removal of entries from flow cache.

Step 6 **cache timeout** {**active** *timeout_value* | **inactive** *timeout_value* | **update** *timeout_value*}

Example:

```
RP/0/RP0/CPU0:router(config-fmm)# cache timeout inactive 120
```

(Optional) Configures the active, inactive, or update flow cache timeout value.

- The default timeout value for the inactive flow cache is 15 seconds.
- The default timeout value for the active flow cache is 1800 seconds.
- The default timeout value for the update flow cache is 1800 seconds.

Note
The **update** *timeout_value* keyword argument is used for permanent caches only. It specifies the timeout value that is used to export entries from permanent caches. In this case, the entries are exported but remain the cache.

Step 7 **exporter** *map_name*

Example:

```
RP/0/RP0/CPU0:router(config-fmm)# exporter expmap-dtxr2
```

Associates an exporter map with a monitor map.

Note
A single flow monitor map can support up to eight exporters.

Step 8 **commit**

Step 9 **exit**

Example:

```
RP/0/RP0/CPU0:router(config-fmm)# exit
```

Exits flow monitor map configuration submode.

Step 10 **exit**

Example:

```
RP/0/RP0/CPU0:router(config)# exit
```

Exits XR Config mode.

Step 11 **show flow monitor-map** *map_name*

Example:

```
RP/0/RP0/CPU0:router#show flow monitor-map fmm-ipv4-dtxr2
Flow Monitor Map : fmm-ipv4-dtxr2
-----------------------------------------------
Id:                 1
RecordMapName:      ipv4-raw
ExportMapName:      expmap-dtxr2
CacheAgingMode:     Normal
CacheMaxEntries:    65535
CacheActiveTout:    60 seconds
CacheInactiveTout: 120 seconds
CacheUpdateTout:    N/A
CacheRateLimit:     2000
```

Displays flow monitor map data.

Example

This example shows how to create a new flow monitor map with name “fmm1”. This flow monitor map references the flow exporter map “fem1,” and sets the flow cache attributes to 10000 cache entries. The active entries from the cache are aged every 30 seconds, while the inactive entries from the cache are aged every 15 seconds. The record map for this monitor map is IPv4:

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm1
RP/0/RP0/CPU0:router(config-fmm)# record ipv4
RP/0/RP0/CPU0:router(config-fmm)# exporter fem1
RP/0/RP0/CPU0:router(config-fmm)# cache entries 10000
RP/0/RP0/CPU0:router(config-fmm)# cache timeout active 30
RP/0/RP0/CPU0:router(config-fmm)# cache timeout inactive 15
RP/0/RP0/CPU0:router(config-fmm)# exit
```

Applying a Monitor Map and a Sampler Map to a Physical Interface

Perform these steps to apply a monitor map and a sampler map to an interface.

Procedure

Step 1 **configure**

Step 2 **interface** *type number*

Example:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/4/0/8
RP/0/RP0/CPU0:router(config-if)#
```

Enters interface configuration mode.

Step 3 **flow** [**ipv4** | **ipv6** | **mpls**] **monitor** *monitor_map* **sampler** *sampler_map* {**ingress**}

Example:

```
RP/0/RP0/CPU0:router(config-if)# flow ipv4 monitor fmm sampler fsm ingress
```

Associates a monitor map and a sampler map with an interface.

Note
Only Ingress mode is supported.

Enter **ipv4** to enable IPV4 NetFlow on the specified interface. Enter **ipv6** to enable IPV6 NetFlow on the specified interface. Enter **mpls** to enable MPLS-aware NetFlow on the specified interface.

Step 4 **commit**

Example

This example shows how to apply the flow monitor “fmm1”and the sampler “fsm1” to the HundredGigE 0/3/0/0 interface in the ingress direction:

```
RP/0/RP0/CPU0:router(config)#interface HundredGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-if)#flow ipv4 monitor fmm1 sampler fsm1 ingress
RP/0/RP0/CPU0:router(config-if)#exit
```

This example shows how to apply the flow monitor “MPLS-IPv6-fmm”and the sampler “FSM” to the HundredGigE 0/3/0/0 interface in the ingress direction:

```
RP/0/RP0/CPU0:router(config)#interface HundredGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-if)# flow mpls monitor MPLS-IPv6-fmm sampler FSM ingress
RP/0/RP0/CPU0:router(config-if)#exit
```

Applying a Monitor Map and a Sampler Map to a Layer 2 Bundle Interface

Perform these steps to apply a monitor map and a sampler map to a Layer 2 bundle interface.

Procedure

Step 1 **configure**

Step 2 **interface** *type number*

Example:

```
RP/0/RP0/CPU0:router(config)# interface bundle-ethernet 1
RP/0/RP0/CPU0:router(config-if)#
```

Enters interface configuration mode.

Step 3 **flow** [**ipv4** | **ipv6** | **mpls**] **monitor** *monitor_map* **sampler** *sampler_map* {**ingress**}

Example:

```
RP/0/RP0/CPU0:router(config-if)# flow ipv4 monitor fmm sampler fsm ingress
```

Associates a monitor map and a sampler map with an interface.

Note
Only Ingress mode is supported.

Enter **ipv4** to enable IPV4 NetFlow on the specified interface. Enter **ipv6** to enable IPV6 NetFlow on the specified interface. Enter **mpls** to enable MPLS-aware NetFlow on the specified interface.

Step 4 **commit**

Example

This example shows how to apply the flow monitor “fmm1”and the sampler “fsm1” to the bundle-ethernet 1 interface in the ingress direction:

```
RP/0/RP0/CPU0:router(config)#interface bundle-ethernet 1
RP/0/RP0/CPU0:router(config-if)#flow ipv4 monitor fmm1 sampler fsm1 ingress
RP/0/RP0/CPU0:router(config-if)#exit
```

This example shows how to apply the flow monitor “MPLS-IPv6-fmm”and the sampler “FSM” to the bundle-ethernet 1 interface in the ingress direction:

```
RP/0/RP0/CPU0:router(config)#interface bundle-ethernet 1
RP/0/RP0/CPU0:router(config-if)# flow mpls monitor MPLS-IPv6-fmm sampler FSM ingress
RP/0/RP0/CPU0:router(config-if)#exit
```

Clearing NetFlow Data

Procedure

Step 1 **clear flow exporter** [*exporter_name*] {**restart** | **statistics**} **location** *node-id*

Example:

```
RP/0/RP0/CPU0:router# clear flow exporter statistics location 0/0/CPU0
```

Clears the flow exporter data.

Specify the **statistics** option to clear exporter statistics. Specify the **restart** option to export all of the templates that are currently configured on the specified node.

Step 2 **clear flow monitor** [*monitor_name*] **cache** [**force-export** | **statistics**] **location** *node-id*}

Example:

```
RP/0/RP0/CPU0:router# clear flow monitor cache force-export location 0/0/CPU0
```

Clears the flow monitor data.

Specify the **statistics** option to clear cache statistics. Specify the **force-export** option to export the data from cache to server first and then clear the entries from cache.

Configure NetFlow Collection of MPLS Packets with IPv6 Fields

The following example shows how to collect MPLS traffic with IPv4 payloads.

```
Router(config)#flow monitor-map MPLS-IPv4-fmm
Router(config-fmm)#record mpls IPv4-fields labels 3
Router(config-fmm)#cache permanent
Router(config-fmm)#exit
Router(config)#interface HundredGigE 0/3/0/0
Router(config-if)#flow mpls monitor MPLS-IPv4-fmm sampler fsm ingress
```

The following example shows how to collect MPLS traffic with IPv6 payloads.

```
Router(config)#flow monitor-map MPLS-IPv6-fmm
Router(config-fmm)# record mpls IPv6-fields labels 3
Router(config-fmm)#cache permanent
```

```
Router(config-fmm)#exit
Router(config)#interface HundredGigE 0/3/0/0
Router(config-if)#flow mpls monitor MPLS-IPv6-fmm sampler fsm ingress
```

The following example shows how to configure the NetFlow monitor to collect MPLS packets with IPv6 fields:

```
Router# config
Router(config)# flow exporter-map exp1
Router(config-fem)#  version v9
Router(config-fem-ver)# options interface-table timeout 300
Router(config-fem-ver)# options sampler-table timeout 300
Router(config-fem-ver)# template data timeout 300
Router(config-fem-ver)# template options timeout 300
Router(config-fem-ver)# exit
Router(config-fem)# transport udp 12515
Router(config-fem)# source Loopback0
Router(config-fem)# destination 170.1.1.11
Router(config-fmm)# exit
Router(config)# flow monitor-map MPLS-IPv6-fmm
Router(config-fmm)# record mpls ipv6-fields labels 3
Router(config-fmm)# exporter exp1
Router(config-fmm)# cache entries 10000
Router(config-fmm)# cache permanent
Router(config-fmm)# exit

Router(config)# sampler-map FSM
Router(config-sm)# random 1 out-of 65535
Router(config-sm)# exit
Router(config)# interface HundredGigE 0/3/0/0
Router(config-if)# flow mpls monitor MPLS-IPv6-fmm sampler FSM ingress
```

The following example shows how to collect MPLS traffic with both IPv6 and IPv4 fields.

```
Router(config)# flow monitor-map MPLS-IPv4-IPv6-fmm
Router(config-fmm)# record mpls IPv4-IPv6-fields labels 3
Router(config-fmm)# cache permanent
Router(config-fmm)# exit
Router(config)# interface HundredGigE 0/3/0/0
Router(config-if)# flow mpls monitor MPLS-IPv4-IPv6-fmm sampler fsm ingress
```

Note Flow records are exported using the Version 9 format.

Running Configuration

```
/* This configuration collects MPLS traffic with IPv4 payloads. */
flow monitor-map MPLS-IPv4-fmm
 record mpls IPv4-fields labels 3
 cache permanent
exit
interface HundredGigE 0/3/0/0
 flow mpls monitor MPLS-IPv4-fmm sampler fsm ingress

/* This configuration collects MPLS traffic with IPv6 payloads. */
flow monitor-map MPLS-IPv6-fmm
 record mpls IPv6-fields labels 3
 cache permanent
```

```
exit
interface HundredGigE 0/3/0/0
 flow mpls monitor MPLS-IPv6-fmm sampler fsm ingress

/* This configuration collects MPLS packets with IPv6 fields */
flow exporter-map exp1
 version v9
  options interface-table timeout 300
  options sampler-table timeout 300
  template data timeout 300
  template options timeout 300
  exit
  transport udp 12515
  source Loopback0
  destination 170.1.1.11
  exit
 flow monitor-map MPLS-IPv6-fmm
 record mpls ipv6-fields labels 3
 exporter exp1
 cache entries 10000
 cache permanent
 exit
sampler-map FSM
random 1 out-of 65535
 exit
 interface HundredGigE 0/3/0/0
 flow mpls monitor MPLS-IPv6-fmm sampler FSM ingress

/* This configuration collects MPLS traffic with both IPv6 and IPv4 fields */
flow monitor-map MPLS-IPv4-IPv6-fmm
 record mpls IPv4-IPv6-fields labels 3
 cache permanent
 exit
 interface HundredGigE 0/3/0/0
 flow mpls monitor MPLS-IPv4-IPv6-fmm sampler fsm ingress
```

Verification

Verify the flow monitor map data.

```
Router# show flow monitor-map MPLS-IPv6-fmm

Flow Monitor Map : MPLS-IPv6-fmm
-----------------------------------------------
Id:                1
RecordMapName:     ipv4-raw
ExportMapName:     expmap-dtxr2
CacheAgingMode:    Normal
CacheMaxEntries:   65535
CacheActiveTout:   60 seconds
CacheInactiveTout: 120 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
```

Verify the exporter map data.

```
Router# show flow exporter-map expmap-dtxr2
Flow Exporter Map : expmap-dtxr2
-----------------------------------------------
Id                   : 1
DestinationIpAddr    : 170.1.1.11
VRFName              : default
```

```
SourceIfName        : Loopback0
SourceIpAddr        : 10.200.58.1
DSCP                : 10
TransportProtocol   : UDP
TransportDestPort   : 12515

Export Version: 9
   Common Template Timeout : 300 seconds
   Options Template Timeout : 300 seconds
   Data Template Timeout : 600 seconds
   Interface-Table Export Timeout : 300 seconds
   Sampler-Table Export Timeout : 0 seconds
   VRF-Table Export Timeout : 0 seconds
```

Verify the netflow cache record for MPLS packet.

```
Router# show flow monitor MPLS-IPv6-fmm cache format  record  location 0/0/CPU0
Thu Feb 25 05:14:11.474 IST
Cache summary for Flow Monitor FNF_MONITOR_MAP_MPLS2:
Cache size:                         256000
Current entries:                         1
Flows added:                            74
Flows not added:                         0
Ager Polls:                           4418
  - Active timeout                      73
  - Inactive timeout                     0
  - Immediate                            0
  - TCP FIN flag                         0
  - Emergency aged                       0
  - Counter wrap aged                    0
  - Total                               73
Periodic export:
  - Counter wrap                         0
  - TCP FIN flag                         0
Flows exported                          73
========== Record number: 1 ==========
LabelType         :       BGP
Prefix/Length     : ::/0
Label1-EXP-S      :        0-0-0
Label2-EXP-S      :    24026-0-1
Label3-EXP-S      :       -
Label4-EXP-S      :       -
Label5-EXP-S      :       -
Label6-EXP-S      :       -
InputInterface    : BE100
OutputInterface   : Hu0/0/0/3.1001
ForwardStatus     : Fwd
FirstSwitched     : 00 06:33:48:047
LastSwitched      : 00 06:33:54:838
ByteCount         : 1002010
PacketCount       : 1033
Dir               : Ing
SamplerID         : 1
IPv6SrcAddr       : 3001:10::2
IPv6DstAddr       : 1001:10::2
IPv6TC            : 0
IPv6FlowLabel     : 7
IPv6OptHdrs       : 0x10
IPV6Prot          : 59
L4SrcPort         : 0
L4DestPort        : 0
L4TCPFlags        : 0
InputVRFID        : default
OutputVRFID       : default
```

Drop Codes on NetFlow

The following table lists supported drop codes on NetFlow, when a node is unable to forward the packets due to various reasons listed here. In such cases, the following drop codes are exported instead of output interface index.

Table 5: Drop Codes on NetFlow

Drop Reason(s)	IPFIX/V9 Code
Unknown	128
ACL Deny	129
Adjacency	132
Bad Header Checksum	134
Bad TTL	137

Additional References

These sections provide references related to interface configuration.

Related Documents

Related Topic	Document Title
Cisco IOS XR interface configuration commands	*Interface and Hardware Component Command Reference for Cisco NCS 5500 and NCS 540 and NCS 560 Series Routers*
Initial system bootup and configuration information for a router using the Cisco IOS XR software.	
Information about user groups and task IDs	*Interface and Hardware Component Command Reference for Cisco NCS 5500 and NCS 540 and NCS 560 Series Routers*
Information about configuring interfaces and other components from a remote Craft Works Interface (CWI) client management application.	Cisco Craft Works Interface User Guide

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.	—

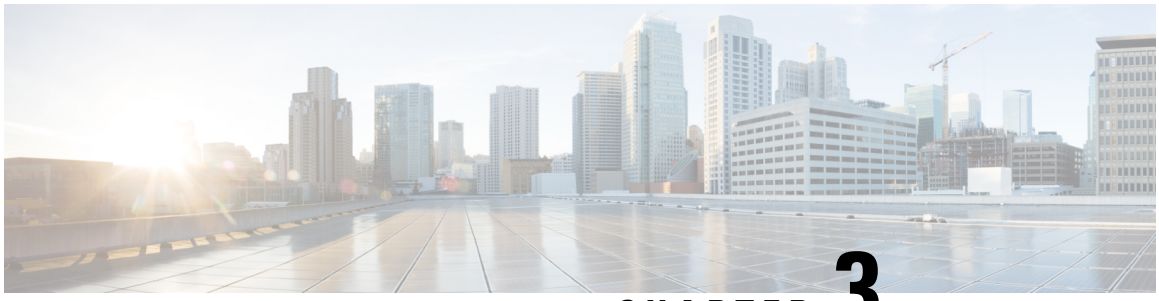
MIBs

MIBs	MIBs Link
—	Text for MIBs: To locate and download MIBs using Cisco IOS XR software, use the MIB Locator found at the Cisco Feature Navigator.

RFCs

RFCs	Title
3954	NetFlow services export protocol Version 9.
7011	IPFIX protocol

Technical Assistance



CHAPTER 3

NetFlow Supported Features

- In-line Modification of Netflow Configuration, on page 27
- Options Template Overview, on page 29
- Flow Filter, on page 31
- IPFIX, on page 34
- NetFlow Configuration Submodes, on page 52
- MPLS Flow Monitor with IPv4 and IPv6 Support, on page 55
- Monitor GTP-U Traffic in 5G Network, on page 58
- Netflow Full Packet Capture, on page 74

In-line Modification of Netflow Configuration

The In-line modification of Netflow configuration enables to add or remove flow attributes of a flow entity that is already applied to an interface.

A flow entity can be a monitor map, exporter map or a sampler map.

Netflow does not support in-line modification of all its configuration items. This table lists flow entries and flow attributes that are in-line modifiable.

Note In-line modification of flow items clears the cache counters. As a result there could be flow accounting mismatch.

Note The In-line modification of Netflow configuration is supported on Cisco IOS XR 64 bit software.

Table 6: In-line Modifiable Flow Entities and Flow Attributes

Flow Entity	Flow Attribute
Monitor map **Note** Any modification to the cache attributes results in resetting of the cache counters. The cache flows are dropped not exported.	cache timeout active *seconds*
	cache timeout inactive *seconds*
	cache timeout update *seconds*
	cache timeout rate-limit *seconds*
	exporter
	cache entries
	cache permanent
	option outphysint | bgstrings **Note** This flow attribute is not supported on Cisco NCS 5500 Router.
Exporter Map **Note** Any modification to an exporter map results in resetting of the exporter counter.	source <*source interface*>
	destination <*destinaiton address*>
	dscp <*dscp_value*>
	version v9 | ipfix
Sampler Map	sampling interval

Restriction

- In-line modification of the **record ipv4** flow attribute is not supported.

Use Case

Consider a netflow configuration as shown below applied on Bundle interface.

```
RP/0/RP1/CPU0:router#show running-config interface bundle-ether 8888
Thu Oct 26 14:17:17.459 UTC
interface Bundle-Ether8888
ipv4 address 192.168.108.1 255.255.255.252
 ipv6 address 192:168:108::1/126
flow ipv6 monitor MONITOR-8k sampler SAMPLER-8k ingress
!
RP/0/RP1/CPU0:router#show running-config flow monitor-map MONITOR-8k
Thu Oct 26 14:17:32.581 UTC
flow monitor-map MONITOR-8k
 record ipv6
 exporter NF-2
 cache timeout update 30
!
```

The Netflow configuration includes:

- flow monitor map—MONITOR-8k: The flow monitor map do not have cache entries configured. Cache entries are the number of entries in the flow cache.
- exporter map—NF-2
- sampler map—SAMPLE-8k

The **cache entries** attribute is in-line modifiable. Let us configure the cache entries, while the flow monitor map is in use:

```
RP/0/RP1/CPU0:router#config
RP/0/RP1/CPU0:router(config)#flow monitor-map MONITOR-8k
RP/0/RP1/CPU0:router(config-fmm)#cache entries 8000
RP/0/RP1/CPU0:router(config-fmm)#commit
Thu Oct 26 14:18:24.625 UTC
RP/0/RP1/CPU0:Oct 26 14:18:24.879 : config[67366]: %MGBL-CONFIG-6-DB_COMMIT : Configuration
 committed by user '<username>'.
Use 'show configuration commit changes 1000000556' to view the changes.  /*configuration
commit is successfull. */
```

The above configuration changes are committed successfully.

Verification

To verify if the monitor map has chache entries of 8000 configured, use the **show flow monitor-map** command for MONITOR-8k map:

```
RP/0/RSP0/CPU0:router# show flow monitor-map MONITOR-8k

Flow Monitor Map : MONITOR-8k
---------------------------------------------------
Id:                1
RecordMapName:     ipv6
ExportMapName:     NF-2
CacheAgingMode:    Permanent
CacheMaxEntries:   8000
CacheActiveTout:   N/A
CacheInactiveTout: N/A
CacheUpdateTout:   30 seconds
```

Options Template Overview

NetFlow version 9 is a template-based version. The templates provide an extensible design to the record format. This feature allows enhancements to NetFlow services without requiring concurrent changes to the basic flow-record format. An options template is a special type of template record that is used to communicate the format of data related to the NetFlow process. Rather than supplying information about IP flows, the options are used to supply metadata about the NetFlow process itself. The sampler options template and the interface options template are different forms of options templates. These two tables are exported by the NetFlow process. The NetFlow process will also export the VRF table.

Sampler Table

The sampler options template consists of sampler tables. Similarly, the interface option templates consist of interface tables. By enabling the options for sampler table and interface table, it becomes easier for the collector to determine the information on data flow.

The sampler table consists of information on the active samplers. It is used by the collector to estimate the sampling rate for each data flow. The sampler table consists of the following information for each sampler:

Field Name	Value
FlowSamplerID	This ID is assigned to the sampler. It is used by the collector to retrieve information about the sampler for a data flow record.
FlowSamplerMode	This field indicates the mode in which the sampling has been performed.
FlowSamplerRandomInterval	This field indicates the rate at which the sampling is performed.
SamplerName	This field indicates the name of the sampler.

Interface Table

The interface table consists of information on interfaces that are being monitored for data flow. By using this information, the collector determines the names of interfaces associated with the data flow. The interface table consists of the following information:

Field Name	Value
ingressInterface	This field indicates the SNMP index assigned to the interface. By matching this value to the Ingress interface in the data flow record, the collector is able to retrieve the name of the interface.
interfaceDescription	This field indicates the name of the interface.

VRF Table

The VRF table consists of mapping of VRF IDs to the VRF names. By using this information, the collector determines the name of the required VRF. The VRF table consists of the following information:

Field Name	Value
ingressVRFID	The identifier of the VRF with the name in the VRF-Name field.
VRF-Name	The VRF name which has the VRFID value ingressVRFID. The value "default" indicates that the interface is not assigned explicitly to a VRF.

The data records contain ingressVRFID as an extra field in each record. The values of these fields are used to lookup the VRF Table to find the VRF names. A value 0 in these fields indicates that the VRF is unknown.

The VRF table is exported at intervals specified by the optional **timeout** keyword that can be configured manually. The default value is 1800 seconds.

Flow Filter

Table 7: Feature History Table

Feature Name	Release Information	Feature Description
Flow Filter on Cisco NC57 Line Cards	Release 7.2.2	With this feature, you can collect user-defined and ACL-filtered NetFlow data that is available in the NetFlow cache and export it to an external interface for processing. Flow filter can be configured on interfaces in ingress direction. This feature introduces the **option filtered** command.

NetFlow provides highly granular per-flow traffic statistics in a Cisco router. The router accumulates NetFlow statistics of all the flows in a NetFlow cache and exports them to an external device for further processing. But in some cases, you might want to gather NetFlow data on only a subset of these flows. The flow filter feature provides the capability to gather NetFlow data on only a specific user-defined subset of flow.

The flow filter feature is configured on interfaces in ingress or egress direction. The flow filter feature uses ACL and QoS bits to filter the NetFlow data; the match criteria is based on five tuple and DSCP bits. The filtered Netflow data is sampled (not all interface flows are sampled) and exported to a collector.

When both security ACL and Netflow filtering ACL are configured on an interface, the security ACL takes precedence over Netflow filtering ACL.

The Flow Filter supports:

- NetFlow v9 and IPFIX export formats.
- Yang data model for dynamic provisioning.

Restrictions

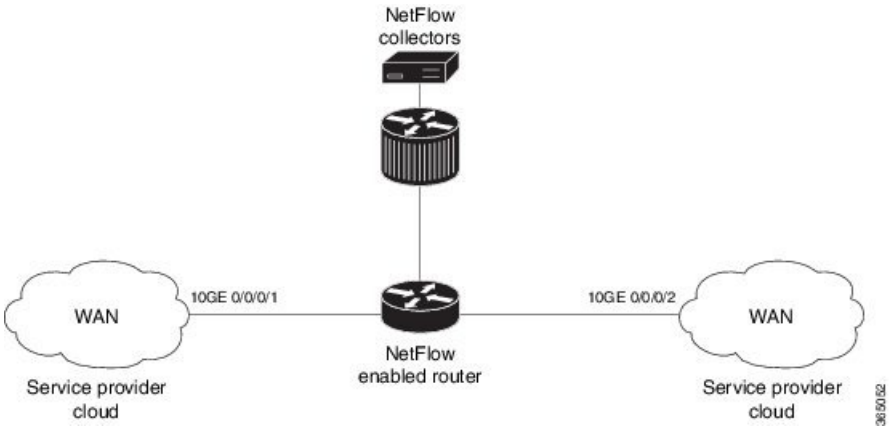
These are the restrictions for the flow filter feature:

- Supported on physical interface, physical subinterface, bundle interface, and bundle subinterface
- Not supported on satellite access interface, ICL interface and clusters.
- MPLS netflow filtering is not supported.

Configuring Flow Filter

Consider SP-PE use case where SP (Service Provide) cloud is connected to the PE (Provider Edge) router through gigabit ethernet.

Figure 2: SP-PE Topology



Configuring NetFlow on PE router involves:

1. Configuring ACL based filter criteria for NetFlow
2. Configuring Monitor map with filter netflow object
3. Configuring Sampler map
4. Configuring Exporter map
5. Applying the NetFlow flow filter ACL configuration and Monitor map to an interface

Configuring ACL based filter criteria for NetFlow

```
ipv4 access-list nf_ex
   10 permit ipv4 192.168.1.1/24 any capture
```

Configuring Monitor map with filter netflow object

```
flow monitor-map fmm1
   record ipv4
    option filtered
   exporter fem1
   cache entries 10000
   cache timeout active 1800
   cache timeout inactive 15
   exit
```

Configuring Sampler map

```
sampler-map fsm1
 random 1 out-of 65535
exit
```

Configuring Exporter map

```
flow exporter-map fem1
 destination 10.1.1.1
 source Loopback 0
 transport udp 1024
 dscp 10
exit
version v9
 template data timeout 600
 options interface-table
exit
```

Applying the NetFlow Flow filter ACL configuration and Monitor map to an interface

```
interface 10GE0/0/0/1
ipv4 access-group nf_ex_ing
flow ipv4 monitor fmm1 sampler fsm1 ingress
exit
```

Verification

Use the **show flow monitor** command to verify the flow filter configuration successfully applied on the PE router:

```
RP/0/RP0/CPU0:router# show flow monitor-map netflow_monitor_map_fl_4

Flow Monitor Map : netflow_monitor_map_fl_4
-----------------------------------------------
Id: 28
RecordMapName: ipv4-raw
ExportMapName: netflow_exporter_map_fl_4
CacheAgingMode: Normal
CacheMaxEntries: 65535
CacheActiveTout: 1800 seconds
CacheInactiveTout: 700 seconds
CacheUpdateTout: N/A
CacheRateLimit: 2000
Options: filtered
HwCacheExists: False
HwCacheInactTout: 50

Flow Monitor :        fmm1
-----------------------------------------------
  Flow definition:    ipv4-raw
  Cache configuration:
    Type:             Normal
    Cache size:                    65535 entries
    Inactive timeout:                 15 seconds
    Active timeout:                 1800 seconds
    Update timeout:                  N/A
    Rate limit:                     2000 entries per second
    Options:                     filtered
```

IPFIX

Internet Protocol Flow Information Export (IPFIX) has been standardized by the Internet Engineering Task Force (IETF) as an export protocol for transmitting NetFlow packets. Building upon NetFlow version 9, IPFIX introduces efficient flow data formatting through templates, ensuring scalability and adaptability to diverse network environments. Utilizing UDP as the transport protocol, IPFIX facilitates the seamless transfer of NetFlow information from exporters to collectors. With native support for IPv6 flow records, the inclusion of optional data fields, and the ability to send data to multiple collectors, IPFIX proves to be a versatile and powerful solution for network administrators, enabling comprehensive traffic analysis, monitoring, and enhanced visibility into network behavior.

Restrictions

These IPFIX features are not supported:

- Variable-length information element in the IPFIX template
- Stream Control Transmission Protocol (SCTP) as the transport protocol

Limitations

- You cannot modify an exporter version of an exporter map that is already applied to an interface. To modify the exporter version, first remove the exporter configuration applied on the interface, later modify the version and apply the configuration to the interface.
- An interface can have three different monitor-maps but all the monitor maps should have the same version for the exporters. There can be different exporters for the three monitor maps but they all need to have the same exporter version either v9 or IPFIX.
- You can only have monitor-maps one of each record type attached to an interface, that is one monitor-map for IPv4 record, one monitor-map for IPv6 record and one for MPLS record. There can be different exporter maps for these three monitor-maps but all the exporter maps should have same exporter version configured, either v9 or IPFIX.
- Multiple sampler-maps can be configured but only two sampler maps can be appled to an interface across the system.

Collect Additional BGP Information Elements for MPLS IPv4 and IPV6 Using IPFIX

Table 8: Feature History Table

Feature Name	Release Information	Feature Description
Collect Additional BGP Information Elements for MPLS IPv4 and IPV6 Using IPFIX	Release 24.1.1	You can now collect insights into how MPLS traffic is flowing through the network, assess the performance of your traffic engineering policies and make informed adjustments, pinpoint where in your MPLS network packets are being misrouted or dropped for swift troubleshooting, and also enable accurate billing for your users' customers because of insights into accurate resource usage. This is made possible because we have enabled the collection of BGP information elements for MPLS IPv4 and IPv6 traffic using IPFIX. This feature modifies the output of the **show flow monitor** command.

You can now monitor and optimize your network more effectively with IPFIX, which enhances the collection of BGP Information Elements (IEs) in IPFIX records. Specifically designed to improve congestion mitigation in core-edge link scenarios, this update introduces support for gathering eight additional BGP fields in IPFIX MPLS IPv4/IPv6 records.

Additionally, two new Information Elements, namely Minimum Time-to-Live (TTL) and Maximum TTL, are recorded. These elements provide information about the minimum Time to Live for a flow and the maximum Time to Live for a flow.

Table 9: Information Elements

IE Field	IE Number
BgpSourceAsNumber	16
BgpDestinationAsNumber	17
BgpNextHopIPv4Address	18
BgpNextHopIPv6Address	63
DestinationIPv4PrefixLength	13
DestinationIPv6PrefixLength	30

IE Field	IE Number
IpNextHopIPv4Address	15
IpNextHopIPv6Address	62
Minimum TTL	52
Maximum TTL	53

IE number, or Information Element Number, is a unique identifier assigned to specific elements within network communication protocols, facilitating standardized interpretation and management. For more information refer IP Flow Information Export (IPFIX) Entities.

Configuration

The following example shows how to collect MPLS traffic with both IPv6 and IPv4 fields.

Configuring Monitor map:

```
Router(config)#flow monitor-map mpls-1
Router(config-fmm)#record mpls ipv4-ipv6-fields
Router(config-fmm)#commit
Router(config-fmm)#exit
```

Configuring Sampler map:

```
Router(config)#sampler-map fsm1
Router(config-sm)#random 1 out-of 4000
Router(config-sm)#commit
Router(config-sm)#exit
```

Apply a Monitor Map and a Sampler Map to a physical interface

```
Router(config)#interface HundredGigE 0/0/0/24
Router(config-if)#flow mpls monitor mpls-1 sampler fsm1 ingress
Router(config-if)#exit
```

Verification

Verify the flow monitor stats statistics using the **show flow monitor cache location command**.

```
Router#show flow monitor mpls-1 cache summary location 0/0/CPU0========== Record number: 1
 ==========
========== Record number: 1 ==========
LabelType        :   Unknown
Prefix/Length    : 20.1.1.0/24
Label1-EXP-S     :    16001-0-1
Label2-EXP-S     :       -
Label3-EXP-S     :       -
Label4-EXP-S     :       -
Label5-EXP-S     :       -
Label6-EXP-S     :       -
InputInterface   : FH0/0/0/1
OutputInterface  : FH0/0/0/0
ForwardStatus    : Fwd
FirstSwitched    : 00 08:28:52:189
LastSwitched     : 00 08:28:57:649
ByteCount        : 2352
PacketCount      : 56
Dir              : Ing
SamplerID        : 1
```

```
IPV4SrcAddr      : 30.1.1.1
IPV4DstAddr      : 20.1.1.1
IPV4TOS          : 0
IPV4Prot         : udp
L4SrcPort        : 2025
L4DestPort       : 2500
L4TCPFlags       : 0
IPV4SrcPrfxLen   : 24
IPV4DstPrfxLen   : 24
BGPNextHopV4     : 192.168.10.10
BGPNextHopV6     : ::
BGPSrcOrigAS     : 2000
BGPDstOrigAS     : 1000
IPV4NextHop      : 192.168.10.10
IPV6NextHop      : ::
MinimumTTL       : 90
MaximumTTL       : 110
InputVRFID       : default
OutputVRFID      : default


========== Record number: 1 ==========
LabelType        :   Unknown
Prefix/Length    : ::/0
Label1-EXP-S     :    16001-0-1
Label2-EXP-S     :       -
Label3-EXP-S     :       -
Label4-EXP-S     :       -
Label5-EXP-S     :       -
Label6-EXP-S     :       -
InputInterface   : FH0/0/0/1
OutputInterface  : FH0/0/0/0
ForwardStatus    : Fwd
FirstSwitched    : 00 08:27:38:692
LastSwitched     : 00 08:27:47:572
ByteCount        : 5580
PacketCount      : 90
Dir              : Ing
SamplerID        : 1
IPv6SrcAddr      : 50::1
IPv6DstAddr      : 40::1
IPv6TC           : 0
IPv6FlowLabel    : 0
IPv6OptHdrs      : 0x0
IPV6Prot         : udp
L4SrcPort        : 2025
L4DestPort       : 2500
L4TCPFlags       : 0
IPV6SrcPrfxLen   : 64
IPV6DstPrfxLen   : 64
BGPNextHopV4     : 0.0.0.0
BGPNextHopV6     : ::ffff:192.168.10.10
BGPSrcOrigAS     : 2000
BGPDstOrigAS     : 1000
IPV4NextHop      : 192.168.10.10
IPV6NextHop      : ::
MinimumTTL       : 195
MaximumTTL       : 205
InputVRFID       : default
OutputVRFID      : default
```

Note When processing the ICMP Layer 4 header, the destination port is determined based on the ICMPv6 message type, instead of being set to zero. This behavior is specific to ICMPv6 and does not apply to ICMP for IPv4.

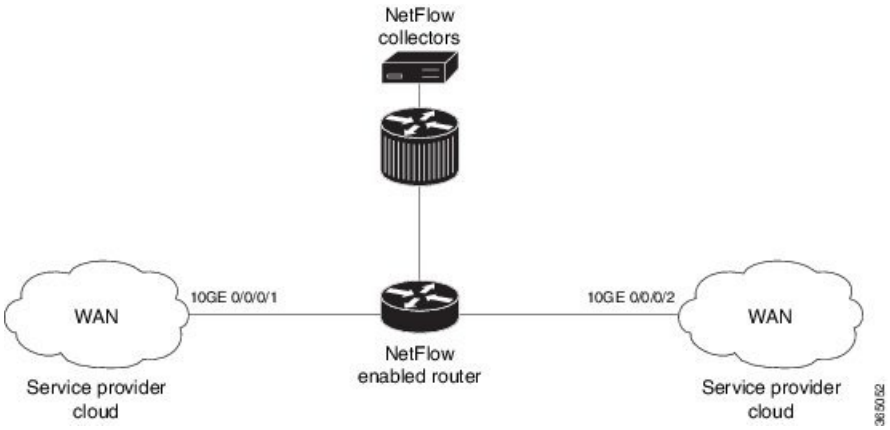
For example, in the following output, the L4DestPort value corresponds to ICMPv6 Msg Type 3 (Time Exceeded). See https://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml#icmpv6-parameters-codes-4

```
IPv6SrcAddr     : 1700::2
IPv6DstAddr     : 1800::2
BGPDstOrigAS    : 0
BGPSrcOrigAS    : 0
BGPNextHopV6    : fcbb:bb00:3::
IPv6TC          : 8
IPv6FlowLabel   : 4
IPv6OptHdrs     : 0x0
IPV6Prot        : icmpv6
MinimumTTL      : 120
MaximumTTL      : 120
L4SrcPort       : 0
L4DestPort      : 3
L4TCPFlags      : 0
IPV6DstPrfxLen  : 64
IPV6SrcPrfxLen  : 128
InputInterface  : Hu0/0/0/1
OutputInterface : Hu0/0/0/0
ForwardStatus   : Fwd
BGPNextHopV4    : 0.0.0.0
IPV4NextHop     : 0.0.0.0
IPV6NextHop     : ::
FirstSwitched   : 04 02:36:55:363
LastSwitched    : 04 02:37:19:963
ByteCount       : 25190
PacketCount     : 229
Dir             : Ing
SamplerID       : 4
SrcMacAddr      : 00:ca:ff:ee:00:01
DstMacAddr      : 04:00:00:07:1d:04
EthType         : 34525
Dot1qPriority   : 0
Dot1qVlanId     : 0
CustVlanId      : 0
InputVRFID      : vrf_1
OutputVRFID     : default
```

Configuring IPFIX

Consider SP-PE use case where SP (Service Provider) cloud is connected to the PE (Provider Edge) router through TenGigabit ethernet.

Figure 3: SP-PE Topology



Configuring NetFlow on PE router involves:

1. Configuring Exporter map with IPFIX as an exporter
2. Configuring Monitor map
3. Configuring Sampler map
4. Applying the Monitor map and Sampler map to an interface

Configuring Exporter map with IPFIX as the exporter version

```
flow exporter-map fem_ipfix
 destination 10.1.1.1
 source Loopback 0
 transport udp 1025
 exit
version ipfix
 template data timeout 600
 options sampler-table
exit
```

Configuring Monitor map

```
flow monitor-map fmm1
   record ipv4
    option filtered
   exporter fem_ipfix
   cache entries 10000
   cache timeout active 1800
   cache timeout inactive 15
   exit
```

Configuring Sampler map

```
sampler-map fsm1
 random 1 out-of 4000  /*Sampling rate supported is 1:4000*/
exit
```

Applying the Monitor map to an interface

Now apply the monitor-map **fmm1** that is configured with an exporter version IPFIX and sampler-map **fsm1** to the 10GE 0/0/0/1 interface in the ingress direction:

```
configure
 interface 10GE0/0/0/1
  flow ipv4 monitor fmm1 sampler fsm1 ingress
 exit
```

Verification

Use the **show flow flow-exporter map** command to verify the exporter version configured is IPFIX:

```
RP/0/RP0/CPU0:router# show flow exporter-map fem_ipfix
Flow Exporter Map : fem_ipfix
-----------------------------------------------
Id                  : 3
Packet-Length       : 1468
DestinationIpAddr   : 10.1.1.1
VRFName             : default
SourceIfName        : Loopback1
SourceIpAddr        : 4.4.0.1
DSCP                : 40
TransportProtocol   : UDP
TransportDestPort   : 9001

Export Version: IPFIX
  Common Template Timeout : 1800 seconds
  Options Template Timeout : 1800 seconds
  Data Template Timeout : 1800 seconds
  Interface-Table Export Timeout : 0 seconds
  Sampler-Table Export Timeout : 0 seconds
  VRF-Table Export Timeout : 0 seconds
```

Exported packets in an IPFIX packet structure are in the form of template set or data set. The first data template is sent when the configuration is activated on the interface.

With constant stream, the flowset data does not change, so data is decoded. Data template is updated in the case of timeout on the template. To change the timeout options in the flow exporter, use the `template options timeout` command:

```
RP/0/RP0/CPU0:router(config)#flow exporter-map ipfix_exp1
RP/0/RP0/CPU0:router(config-fem)#version ipfix
RP/0/RP0/CPU0:router(config-fem-ver)#template options
RP/0/RP0/CPU0:TU-PE3(config-fem-ver)#template options timeout
RP/0/RP0/CPU0:TU-PE3(config-fem-ver)#template options timeout 30

RP/0/RP0/CPU0:router# show flow exporter-map ipfix_exp1
version ipfix

  template data timeout 30
!
dscp 40
transport udp 9001
source Loopback0
destination 10.127.59.86
```

IPFIX Enablement for SRv6 and Services over SRv6 Core

Table 10: Feature History Table

Feature Name	Release Information	Description
IPFIX Enablement for SRv6 and Services over SRv6 Core	Release 7.8.1	This feature provides improved information elements about SRv6 IP traffic flows recorded by IPFIX from the network devices. The following sub-menus are introduced for this command: The **record ipv6** command is modified to support a new optional keyword, **srv6**. For more information, see: • record ipv6 • show flow monitor-map

Feature Name	Release Information	Description
Simultaneous L2 and L3 Flow Monitoring using IPFIX	Release 7.10.1	Introduced in this release on: NCS 5500 fixed port routers NCS 5500 modular routers (NCS 5500 line cards) This feature introduces support for simultaneous L2 and L3 flow monitoring. Now, you can configure IP Flow Information Export (IPFIX) to actively monitor and record end-to-end L2 and L3 flow information elements from network devices. Previously, only L2 or L3 flow could be monitored at a time. This feature introduces these changes: **CLI:** The following sub-menus are introduced for these commands: • The **record ipv4** command is modified to support a new optional keyword, **l2-l3** • The **record ipv6** command is modified to support a new optional keyword, **l2-l3** **YANG Data Model:** • New XPaths for `Cisco-IOS-XR-UM-flow-cfg.yang` (see GitHub, YANG Data Models Navigator)

When migrating from traditional IP and MPLS networks to SRv6-based networks, there is a need for information elements specific to SRv6 traffic flow. To address this, we have introduced the **srv6** keyword to the **ipv6** command. By utilizing this keyword, you can now access SRv6 flow information that is recorded by IPFIX from the network devices.

Restriction and Limitation

1. IPFIX with multiple SRH is not supported in IOS XR software version 7.10.1
2. SRv6 NetFlow is not supported on subinterfaces of decap nodes, including both L2VPN and L3VPN scenarios. To address this limitation, you can apply NetFlow on the main interface instead, which can capture traffic over the underlying subinterface and populate the record. However, please be aware that in the NetFlow record, the input ifhandle will be associated with the main interface only.

Configuration

From Cisco IOS-XR Release 7.8.1, a new optional keyword, `srv6` is introduced for the `record ipv6` option. See the following example:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config-fem)# flow monitor-map MON-MAP-v6
RP/0/RP0/CPU0:router(config-fmm)# record ipv6 srv6
RP/0/RP0/CPU0:router(config-fmm)# exporter EXP
RP/0/RP0/CPU0:router(config-fmm)# cache timeout inactive 5
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# sampler-map SAMP
RP/0/RP0/CPU0:router(config-fmm)# random 1 out-of 1000
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# interface GigabitEthernet0/1/0/0
RP/0/RP0/CPU0:router(config-fmm)# ipv6 address 2002:1::1/64
RP/0/RP0/CPU0:router(config-fmm)# flow ipv6 monitor M1 sampler SAMP ingres
```

This example shows how to display SRv6 monitor-map data for a specific flow:

```
RP/0/RP0/CPU0:router# show flow monitor-map MON

Flow Monitor Map : MON
---------------------------------------------------
Id:                1
RecordMapName:     srv6
ExportMapName:     EXP
CacheAgingMode:    Normal
CacheMaxEntries:   65535
CacheActiveTout:   101 seconds
CacheInactiveTout: 15 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

From Cisco IOS-XR Release 7.10.1, a new optional keyword, `l2-l3` is introduced for the `record ipv4` and `record ipv6` option. By utilizing this keyword, you can now access end-to-end L2 and L3 flow information that is recorded by IPFIX from the network devices. See the following example:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config-fem)# flow monitor-map M-IPv4
RP/0/RP0/CPU0:router(config-fmm)# record ipv4 l2-l3
RP/0/RP0/CPU0:router(config-fmm)# exporter EXP-ipfix
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# flow monitor-map M-IPv6
RP/0/RP0/CPU0:router(config-fmm)# record ipv6 l2-l3
RP/0/RP0/CPU0:router(config-fmm)# exporter EXP-ipfix
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# sampler-map SAMP
RP/0/RP0/CPU0:router(config-fmm)# random 1 out-of 1000
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# interface GigabitEthernet0/1/0/0
RP/0/RP0/CPU0:router(config-fmm)# description CE-PE Interface
RP/0/RP0/CPU0:router(config-fmm)# ipv4 address 1.1.1.1 255.255.255.0
RP/0/RP0/CPU0:router(config-fmm)# ipv6 address 2001:DB8:c18:1::/64
RP/0/RP0/CPU0:router(config-fmm)# flow ipv4 monitor M-IPv4 sampler SAMP ingres
```

```
RP/0/RP0/CPU0:router(config-fmm)# flow ipv6 monitor M-IPv6 sampler SAMP ingress
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router
```

This example shows how to display IPv4 monitor-map data for a specific flow:

```
RP/0/RP0/CPU0:router# show run flow monitor-map

flow monitor-map M-IPv4
 record ipv4 l2-l3
 exporter EXP
!
flow monitor-map M-IPv6
 record ipv6 l2-l3
 exporter EXP
!
```

This example shows how to display l2-l3 monitor-map data for IPv4 specific flow:

```
RP/0/RP0/CPU0:router# show flow monitor-map M-IPv4

Flow Monitor Map : M-IPv4
-----------------------------------------------
Id:                3
RecordMapName:     ipv4-l2-l3
ExportMapName:     EXP
CacheAgingMode:    Normal
CacheMaxEntries:   65535
CacheActiveTout:   1800 seconds
CacheInactiveTout: 15 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

This example shows how to display l2-l3 monitor-map data for IPv6 specific flow:

```
RP/0/RP0/CPU0:router# show flow monitor-map M-IPv6

Flow Monitor Map : M-IPv6
-----------------------------------------------
Id:                4
RecordMapName:     ipv6-l2-l3
ExportMapName:     EXP
CacheAgingMode:    Normal
CacheMaxEntries:   65535
CacheActiveTout:   1800 seconds
CacheInactiveTout: 15 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

This example shows the complete recorded data for SRv6 L2 services :

```
RP/0/RP0/CPU0:router# show flow monitor M-IPv6 location 0/0/CPU0

Cache summary for Flow Monitor M1:
Cache size:                         65535
Current entries:                        3
```

```
Flows added:                            4
Flows not added:                        0
Ager Polls:                         68143
  - Active timeout                      0
  - Inactive timeout                    1
  - Immediate                           0
  - TCP FIN flag                        0
  - Emergency aged                      0
  - Counter wrap aged                   0
  - Total                               1
Periodic export:
  - Counter wrap                        0
  - TCP FIN flag                        0
Flows exported                          1

========== Record number: 1 ==========
IPv6SrcAddr           : 2::2
IPv6DstAddr           : bbbb:bc00:88:e000::
BGPDstOrigAS          : 0
BGPSrcOrigAS          : 0
BGPNextHopV6          : fe80::232:17ff:fe7e:1ce1
IPv6TC                : 0
IPv6FlowLabel         : 50686
IPv6OptHdrs           : 0x0
IPV6Prot              : 143
L4SrcPort             : 0
L4DestPort            : 0
L4TCPFlags            : 0
IPV6DstPrfxLen        : 48
IPV6SrcPrfxLen        : 128
InputInterface        : Hu0/0/0/10
OutputInterface       : BE111.1
ForwardStatus         : Fwd
FirstSwitched         : 01 18:51:25:797
LastSwitched          : 01 18:51:25:797
ByteCount             : 61004304
PacketCount           : 113814
Dir                   : Ing
SamplerID             : 1
InputVRFID            : default
OutputVRFID           : default
InnerIPV4SrcAddr      : 0.0.0.0
InnerIPV4DstAddr      : 0.0.0.0
InnerIPv6SrcAddr      : ::
InnerIPv6DstAddr      : ::
InnerL4SrcPort        : 0
InnerL4DestPort       : 0
SrcMacAddr            : 00:0c:29:0e:d8:32
DstMacAddr            : 00:0c:29:0e:d8:3c
EthType               : 2048
Dot1qPriority         : 0
Dot1qVlanId           : 2001
RecordType            : SRv6 L2 Service Record
SRHFlags              : 0x0
SRHTags               : 0x0
SRHSegmentsLeft       : 0
SRHNumSegments        : 0
```

This example shows the complete recorded data for IPv6 L2-L3 services :

```
RP/0/RP0/CPU0:router# show flow monitor M-IPv6 location 0/0/CPU0
```

```
RP/0/RP0/CPU0:router# show flow monitor MON-MAP-v6 location 0/0/CPU0
Thu Apr 28 11:36:47.622 IST
…
========== Record number: 1 ==========
IPv6SrcAddr           : 151:1::1
IPv6DstAddr           : ff02::1:ff00:2
BGPDstOrigAS          : 0
BGPSrcOrigAS          : 0
BGPNextHopV6          : ::
IPv6TC                : 224
IPv6FlowLabel         : 0
IPv6OptHdrs           : 0x0
IPV6Prot              : icmpv6
MinimumTTL            : 255
MaximumTTL            : 255
L4SrcPort             : 0
L4DestPort            : 135
L4TCPFlags            : 0
IPV6DstPrfxLen        : 0
IPV6SrcPrfxLen        : 0
InputInterface        : BE999.1
OutputInterface       : 0
ForwardStatus         : FwdNoFrag
FirstSwitched         : 01 18:51:25:797
LastSwitched          : 01 18:51:25:797
ByteCount             : 104
PacketCount           : 1
Dir                   : Ing
SamplerID             : 1
InputVRFID            : default
OutputVRFID           : default
SrcMacAddr            : 00:0c:29:0e:d8:32
DstMacAddr            : 00:0c:29:0e:d8:3c
EthType               : 2048
Dot1qPriority         : 0
Dot1qVlanId           : 100
CustVlanId            : 200
```

BGP community and AS path information elements for IPFIX

BGP community and AS path information elements are IPFIX (NetFlow v10) data elements that

- enable tagging of network flows with BGP community and AS path values
- allow users to correlate flow records with BGP path attributes for both communities and autonomous system numbers, and
- support enhanced flow analysis, troubleshooting, and policy verification.

BGP communities and AS paths

BGP communities are mechanisms that tag routes with additional information, making it easier for network operators to manage routing policies and analyze routing decisions based on BGP attributes.

AS path represents the sequence of autonomous systems a route has traversed, providing insight into the journey of a route across multiple networks and enabling more informed routing decisions.

Table 11: Feature History Table

Exported attributes information

The IPFIX (NetFlow v10) export supports two primary BGP attributes:

- bgpDestinationCommunityList (IE485): Exports a list of BGP community values associated with the destination of each flow.
- bgpDestinationAsPathList (IE512): Exports the sequence of autonomous systems (AS path) associated with the destination prefix of each flow.

Guidelines for exporting BGP and AS attributes using IPFIX

Requirements for exporting BGP and AS attributes using IPFIX

- Set the exporter map to IPFIX version to export BGP attributes.
- Exported list is limited to 32 elements. If a list contains more than 32 elements, it will be truncated, and only the first 32 elements will be exported.
- Verify BGP attribute export by checking the exported packets at the collector because show commands do not display these changes.

Caution for exporting BGP and AS attributes using IPFIX

- This export feature is specific to NetFlow v10 (IPFIX).

Configure BGP community and AS path export for IPFIX

Use this procedure when you need to correlate IPFIX traffic records with BGP routing attributes. This is required for advanced traffic analysis, troubleshooting, or policy enforcement use cases

Before you begin

Before starting this procedure, make sure you have completed the necessary prerequisites.

- Ensure you are running release 25.3.1 or later.
- Confirm that your platform supports IPFIX and the relevant flow monitor and exporter configurations.

Follow these steps to configure BGP community and AS path export for IPFIX:

Procedure

Step 1 Configure the Flow Monitor Map for IPv4 Recording and Export of BGP Community and AS Path Attributes.

```
Router# configure
Router(config)# flow monitor-map fmm-test
Router(config-fmm)# record ipv4
Router(config-fmm)# option bgpattr
Router(config-fmm)# exporter fem-test
Router(config-fmm)# exit
```

Step 2 Verify that your exporter map specifies `version ipfix`.

```
Router(config)# flow exporter-map fem-test
Router(config-fem)# version ipfix
Router(config-fem)# transport udp 2055
Router(config-fem)# source-address 10.0.0.2
Router(config-fem)# destination 10.0.0.1
Router(config-fem)# exit
```

Step 3 Configure the sample map.

```
Router(config)# sampler-map fsm-test
Router(config-samplermap)# random 1 out-of 8000
Router(config-samplermap)# exit
```

Step 4 Apply the flow monitor map to the required interface.

```
Router(config)# interface FourHundredGigE0/0/0/17
Router(config-if)# flow ipv4 monitor fmm-test sampler fsm-test ingress
Router(config-if)# end
```

Step 5 Verify BGP community and AS path attribute export.

```
Router# show run flow monitor-map fmm-test
Fri Aug 29 05:40:58.443 UTC
flow monitor-map fmm-test
 record ipv4
 option bgpattr
 exporter fem-test
!
```

BGP community and AS path information elements are included in exported IPFIX flow records.

IP Flow Information Export (IPFIX) 315

Internet Protocol Flow Information Export (IPFIX) is an IETF standard export protocol (RFC 7011) for sending IP flow information. Cisco NCS 5500 Router supports IPFIX 315 format to export flow information. IPFIX 315 format facilitates sending ‘n’ octets frame information starting from ethernet header till transport header of the traffic flow over the network. IPFIX 315 supports sending variable size packet record with variable payload information such as IPv4, IPv6, MPLS, and Nested packets like OuterIP-GRE-InnerIP and so on. The process includes sampling and exporting the traffic flow information. Along with the ethernet frame information, IPFIX 315 format exports information of incoming and outgoing interface of the sampled packet.

Use **hw-module profile netflow ipfix315 location** < *linecard location* > command to enable IPFIX 315.

The information of the packets flowing through a device is used for variety of purpose including network monitoring, capacity planning, traffic management, and so on,

Note Cisco NCS 5500 Router does not support Netflow version 9 format to export flow information.

Sampling and Exporting Information

You must configure a sampling map to sample the traffic flow information. The sampler map specifies the rate at which packets (one out of n packets) are sampled. The minimun sampling rate is 1 out of 32,000 packets.

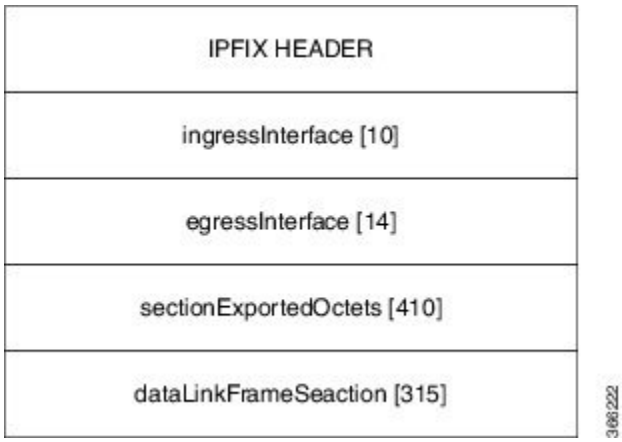
Not all packets flowing through a device are exported; packets selected as per sampling rate are considered for exporting.

You must configure a sampling map to sample the traffic flow information. The sampler map specifies the rate at which packets (one out of n packets) are sampled.

- The size of the exported packet is determined by the sampled packet size from the outermost Layer 2 header.
- If the sampled packet size is less than 160 bytes, the actual packet size is exported.
- If the sampled packet size is 160 bytes or more, the exported packet size is capped at 160 bytes.
- The exported packet includes data up to the Layer 4 header if present; however, if the Layer 4 header is not found within the first 160 bytes, the export size is limited to 160 bytes.

The below figure *IPFIX 315 Export Packet Format* shows exported packet information.

Figure 4: IPFIX 315 Export Packet Format



A special cache type called Immediate Aging is used while exporting the packets. Immediate Aging ensures that the flows are exported as soon as they are added to the cache. Use the command **cache immediate** in flow monitor map configuration to enable Immediate Aging cache type.

IPFIX 315 Implementation Considerations

Here are few key points to consider before implementing IPFIX 315:

- Supported only in ingress direction.
- Supported on main interface only. The traffic on all sub-interfaces under the main interface is exported. This applies to releases up to and including IOS-XR software release 7.10.x.
- Sampling rate for bundles is per member-link and not per bundle interface.
- The outgoing interface information may not be correct incase of packets that are multicasted or broadcasted on multiple ports.
- The incoming and outgoing interface will have information of main interface and not the sub-interface even if the packet is routed via sub-interface. Incase of bundles it will point to bundle main interface.
- IPFIX 315 is not supported on BVI interface.

- Sampling and exporting of the control packets is not supported.
- When you configure **ipfix315-enable**, then you must configure all the ports on that LC with `datalinkframesection` flow.
- When the HQoS profile is enabled, Netflow does not give correct Output Interface. DSP is unique for each sub-interface.
- Netflow on the L2 interface assumes IPv4/IPv6/MPLS traffic, and if the traffic is purely L2 based, then the system ignores that traffic.
- You must remove all v9 configurations before reloading an LC. Else, with the existing v9 configurations on LC reload, you might encounter a few configuration apply error. Or, flow might be seen on an interface even when apply on interface has failed.

Configuring IPFIX 315

Configuring IPFIX 315 involves:

1. Configuring Exporter map
2. Configuring Monitor map
3. Configuring Sampler map
4. Enabling IPFIX 315 on a line card
5. Applying the Monitor map and Sampler map to an interface

Configuring Exporter map

```
flow exporter-map ipfix_exp
 version ipfix
 !
 dscp 40
 transport udp 9001
 source Loopback1
 destination 100.10.1.159
!
```

Note For **options** command and its configurations in Exporter Map, see options.

Configuring Monitor map

```
flow monitor-map ipfix_mon
 record datalinksectiondump
 exporter ipfix_exp
 cache immediate
 cache entries 1000000
 cache timeout rate-limit 1000000
!
```

Configuring Sampler map

```
sampler-map ipfix_sm
 random 1 out-of 32000
!
```

Note The default cache size is 65535, hence you can configure sampling rate as 1 out of 65535 packets. However the recommended sampling rate is 1 out of 32000 packets.

Enabling IPFIX 315 on a line card

```
(config)# hw-module profile netflow ipfix315-enable location 0/0/CPU0
```

You should reload the LC for the changes to take effect.

Applying the Monitor map to an interface

```
interface HundredGigE 0/0/0/18
        flow datalinkframesection monitor ipfix_mon sampler ipfix_sm ingress
```

Verification

Use the **show flow platform producer statistics location** command to display the IPFIX 315 ingress packets flow statistics:

```
RP/0/RP0/CPU0#show flow platform producer statistics location 0/0/CPU0
Netflow Platform Producer Counters:
IPv4 Ingress Packets:                        0
IPv4 Egress Packets:                         0
IPv6 Ingress Packets:                        0
IPv6 Egress Packets:                         0
MPLS Ingress Packets:                        0
MPLS Egress Packets:                         0
IPFIX315 Ingress Packets:               630478
IPFIX315 Egress Packets:                     0
Drops (no space):                            0
Drops (other):                               0
Unknown Ingress Packets:                     0
Unknown Egress Packets:                      0
Worker waiting:                           2443
```

Use the **show flow monitor <monitor-map> cache location** command to check the flow monitor stats. In this example flow statistics for *ipfix_mon* monitor map are displayed:

```
RP/0/RP0/CPU0#show flow monitor ipfix_mon cache location 0/0/CPU0

Cache summary for Flow Monitor ipfix_mon:
Cache size:                         65535
Current entries:                        0
Flows added:                        50399
Flows not added:                        0
Ager Polls:                          2784
  - Active timeout                      0
  - Inactive timeout                    0
  - Immediate                       50399 /*cache type immediate*/
  - TCP FIN flag                        0
  - Emergency aged                      0
  - Counter wrap aged                   0
  - Total                           50399
```

```
Periodic export:
  - Counter wrap                        0
  - TCP FIN flag                        0
Flows exported                      50399


Matching entries:                       0
```

Above example shows that there were 50399 flows added to the cache and exported.

NetFlow Configuration Submodes

In Cisco IOS XR Software, NetFlow map configuration takes place in map-specific submodes. Cisco IOS XR Software supports these NetFlow map configuration submodes:



Note The Cisco IOS XR Software allows you to issue most commands available under submodes as one single command string from mode. For example, you can issue the **record ipv4** command from the flow monitor map configuration submode as follows:

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm
RP/0/RP0/CPU0:router(config-fmm)# record ipv4
```

Alternatively, you can issue the same command from global configuration mode, as shown in the following example:

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm record ipv4
```

Flow Monitor Map Configuration Submode

When you issue the **flow monitor-map** *map_name* command in mode, the CLI prompt changes to "config-fmm," indicating that you have entered the flow monitor map configuration submode.

In this sample output, the question mark (?) online help function displays all the commands available under the flow monitor map configuration submode:

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm

RP/0/RP0/CPU0:router(config-fmm)# ?

cache     Specify flow cache attributes
commit    Commit the configuration changes to running
describe  Describe a command without taking real actions
do        Run an exec command
exit      Exit from this submode
exporter  Specify flow exporter map name
no        Negate a command or set its defaults
record    Specify a flow record map name
show      Show contents of configuration
```

Flow Exporter Map Configuration Submode

Table 12: Feature History Table

Feature Name	Release Information	Description
sFlow Agent Address Assignment	Release 7.10.1	Introduced in this release on: NCS 5500 fixed port routers; NCS 5700 fixed port routers; NCS 5500 modular routers (NCS 5500 line cards; NCS 5700 line cards [Mode: Compatibility; Native]) You can now monitor traffic from a specific source by configuring the sFlow agent ID with the specific IPv4 or IPv6 address. Upon configuration, you can determine the source of the sFlow data. Earlier, by default, the sFlow agent ID had the source address of the sFlow export packet. The feature introduces these changes: **CLI** New Command: • **router-id** Modified Command: • The **show flow exporter-map** command is modified to display flow exporter map with router-id information. **YANG Data Model** • New XPaths for `openconfig-sampling-sflow.yang` (see GitHub, YANG Data Models Navigator)

When you issue the **flow exporter-map** *fem-name* command in mode, the command-line interface (CLI) prompt changes to "config-fem," indicating that you have entered the flow exporter map configuration submode.

In this sample output, the question mark (?) online help function displays all the commands available under the flow exporter map configuration submode:

```
RP/0/RP0/CPU0:router(config)# flow exporter-map fem

RP/0/RP0/CPU0:router(config-fem)# ?

clear        Clear the uncommitted configuration
clear        Clear the configuration
commit       Commit the configuration changes to running
describe     Describe a command without taking real actions
destination  Export destination configuration
do           Run an exec command
dscp         Specify DSCP value for export packets
exit         Exit from this submode
no           Negate a command or set its defaults
pwd          Commands used to reach current submode
root         Exit to the global configuration mode
router-id    router-id or agent-id configuration
show         Show contents of configuration
source       Source interface
transport    Specify the transport protocol for export packets
version      Specify export version parameters
```

Note If you enter the **version** command, you enter the flow exporter map version configuration submode.

Note A single flow monitor map can support up to eight exporters.

Flow Exporter Map Version Configuration Submode

When you issue the **version v9** command in the flow exporter map configuration submode, the CLI prompt changes to "config-fem-ver," indicating that you have entered the flow exporter map version configuration submode.

In this sample output, the question mark (?) online help function displays all the commands available under the flow exporter map version configuration submode:

```
RP/0/RP0/CPU0:router(config-fem)# version v9

RP/0/RP0/CPU0:router(config-fem-ver)# ?

commit    Commit the configuration changes to running
describe  Describe a command without taking real actions
do        Run an exec command
exit      Exit from this submode
no        Negate a command or set its defaults
options   Specify export of options template
show      Show contents of configuration
template  Specify template export parameters
```

Sampler Map Configuration Submode

When you issue the **sampler-map** *map_name* command in mode, the CLI prompt changes to "config-sm," indicating that you have entered the sampler map configuration submode.

In this sample output, the question mark (?) online help function displays all the commands available under the sampler map configuration submode:

```
RP/0/RP0/CPU0:router(config)# sampler-map fmm

RP/0/RP0/CPU0:router(config-sm)# ?
clear     Clear the uncommitted configuration
clear     Clear the configuration
commit    Commit the configuration changes to running
describe  Describe a command without taking real actions
do        Run an exec command
exit      Exit from this submode
no        Negate a command or set its defaults
pwd       Commands used to reach current submode
random    Use random mode for sampling packets
root      Exit to the global configuration mode
show      Show contents of configuration
```

Enabling the NetFlow BGP Data Export Function

Use the **bgp attribute-download** command to enable NetFlow BGP routing attribute collection. The routing attributes are then exported. When no routing attributes are collected, zeroes (0) are exported.

When BGP attribute download is enabled, BGP downloads the attribute information for prefixes (community, extended community, and as-path) to the Routing Information Base (RIB) and Forwarding Information Base (FIB). This enables FIB to associate the prefixes with attributes and send the NetFlow statistics along with the associated attributes.

MPLS Flow Monitor with IPv4 and IPv6 Support

Cisco IOS XR Software supports the NetFlow collection of MPLS packets. It also supports the NetFlow collection of MPLS packets carrying IPv4, IPv6, or both IPv4 and IPv6 payloads.

MPLS Cache Reorganization to Support Both IPv4 and IPv6

In Cisco IOS XR Software, at a time, you can have only one MPLS flow monitor running on an interface. If you apply an additional MPLS flow monitor to the interface, the new flow monitor overwrites the existing one.

You can configure the MPLS flow monitor to collect IPv4 fields, IPv6 fields, or IPv4-IPv6 fields. IPv4-IPv6 configuration collects both IPv4 and IPv6 addresses using one MPLS flow monitor. IPv4 configuration collects only IPv4 addresses. IPv6 configuration collects only IPv6 addresses.

The MPLS flow monitor supports up to 1,000,000 cache entries. NetFlow entries include these types of fields:

- IPv4 fields
- IPv6 fields

- MPLS with IPv4 fields
- MPLS with IPv6 fields

The maximum number of bytes per NetFlow cache entry is as follows:

- IPv4–88 bytes per entry
- IPv6–108 bytes per entry
- MPLS with IPv4 fields–108 bytes per entry
- MPLS with IPv6 fields–128 bytes per entry

Note The different types of NetFlow entries are stored in separate caches. Consequently, the number of NetFlow entries on a line card can significantly impact the amount of available memory on the line card. Also, even though the sampling rate for IPv6 is the same as the sampling rate for IPv4, the CPU utilization for IPv6 is higher due to the longer keys used by the IPv6 fields.

MPLS Packets with IPv6 Flows

Table 13: Feature History Table

Feature Name	Release Information	Feature Description
MPLS top label type 4 for BGP Labeled Unicast traffic	Release 7.4.1	This feature is an enhancement to how Netflow MPLS records are verified. This feature allows the user to analyze the traffic types by providing more visibility on the granularity of the information. This feature helps you to monitor the traffic data. This feature introduces the new MPLS label type BGP. This label type is a field in the MPLS label that identifies the control protocol which allocates the top-of-stack label. MPLS label types enable verification of Netflow MPLS records.

The collection of IPv6 flows in MPLS packets is an option. The CPU uses 128 bytes for each IPv6 field. IPv6 flows may contain these types of information:

- Source IP address
- Destination IP address
- Traffic class value
- Layer 4 protocol number
- Layer 4 source port number
- Layer 4 destination port number
- Flow ID

- Header option mask

To collect the IPv6 fields in MPLS packets, you must activate the MPLS record type, ipv6-fields by running the **record mpls ipv6-fields** command. You can also specify the number of labels to be used for aggregation with this command.

Top label type 4 for BGP Labeled Unicast traffic

MPLS packets sampled by the netflow monitor export the label type based on the topmost label type in the netflow cache record. When the topmost record is an explicit NULL, the succeeding label type is accounted.

The Top label type 4 for BGP Labeled Unicast traffic feature is an enhancement to how netflow MPLS records are verified. MPLS label type value 4, which indicates any label associated with BGP or BGP routing, is supported starting from Release 7.4.1. Earlier to this release, the label type was exported as 0, indicating unknown.

This feature provides the user with additional support for analysis of traffic types by providing more visibilty on the granularity of information. This feature provides clearer perspective on data monitoring.

Table 14: Netflow cache record of MPLS label types and values

Value	Description	Supported available
0	Unknown: The MPLS label type is not known	Yes
1	TE-MIDPT: Any TE tunnel mid-point or tail label	No
2	Pseudowire: Any PWE3 or Cisco AToM-based label	No
3	VPN: Any label associated with VPN	No
4	BGP: Any label associated with BGP or BGP routing	Yes
5	LDP: Any label associated with dynamically assigned labels using LDP	Yes
6-255	Unassigned	No

Only label types 0, 4, and 5 are supported. Labels that are not advertised from LDP and BGP are exported as 0, indicating the value as "unknown".

Monitor GTP-U Traffic in 5G Network

Table 15: Feature History Table

Feature Name	Release Information	Feature Description
Monitor GTP-U Traffic in 5G Network	Release 24.2.1	NCS 5500 fixed port routers; NCS 5700 fixed port routers; NCS 5500 modular routers (NCS 5500 line cards; NCS 5700 line cards [Mode: Compatibility; Native]) You now get a comprehensive view of your 5G network's performance and gain detailed insights into slice utilization, deployed QoS policies, and their impact on traffic. This includes verifying deployed QoS policies, assessing 5G slice mechanisms, and tracking GTP-U endpoints for specific applications. This feature specifically applies to 5G network slicing when the GTP User Plane carries data within the core network and to the radio access network. This is achieved by exporting GTP-U related Information Elements using Netflow and IPFIX records to collectors for analysis. This feature introduces these changes: **CLI:** • The **gtp** keyword is introduced in the **record ipv4** and **record ipv6** commands.

Cisco 8000 routers introduces the capability to monitor the performance of GTP-U traffic in 5G networks. This feature utilizes Netflow and IPFIX to collect and analyze traffic data, offering valuable insights into network performance and facilitating effective management of 5G network traffic.

Starting from IOS-XR software release 24.2.1, three new GTP-U related information elements can be gathered in Netflow and IPFIX records for both IPv4 and IPv6 traffic. This advancement allows administrators to optimize the performance and security of their 5G networks.

The newly introduced information elements are as follows:

IE Field	IE Number
GTP_TEID	507
GTP_QFI	509
GTP_SESS_DIR	510

IE number, or Information Element Number, is a unique identifier assigned to specific elements within network communication protocols, facilitating standardized interpretation and management. For more information, refer IP Flow Information Export (IPFIX) Entities.

Benefits of GTP-U Traffic Monitoring

The following are some of the key benefits of enabling GTP-U traffic monitoring on your router.

- Monitor Network Slicing: 5G network slicing enables the creation of dedicated virtual networks with specific functionalities. By exporting GTP traffic records, you can conduct detailed analysis of the traffic within each slice, ensuring optimal performance and resource allocation.
- Flexible Deployment: GTP-U monitoring can be implemented on any network node where the outermost traffic encapsulation utilizes the GTP protocol. This capability can be activated to monitor traffic at various strategic points across the network infrastructure.
- IPv6 Support for 5G Deployments: With the expansion of 5G networks, there's an increasing use of IPv6, especially in scenarios where 5G base stations (gNodeBs) connect to User Plane Functions (UPFs) using IPv6. This feature ensures that flow records for such IPv6 GTP-U traffic can be captured and exported effectively.

GTP-U Traffic Record Templates

This section provides you with all the record template options available for monitoring GTP-U traffic.

IPv4-GTP-IPv4 Record

This record captures GTP-U traffic details between IPv4 interfaces, essential for monitoring and optimizing IPv4 5G network performance.

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
46	256	V9_ETH_TYPE	2	256	V9_ETH_TYPE	2
47	243	V9_DOT1Q_VLAN_ID	2	243	V9_DOT1Q_VLAN_ID	2
48	245	V9_DOT1Q_CUST_VLAN_ID	2	245	V9_DOT1Q_CUST_VLAN_ID	2
49	244	V9_DOT1Q_PRIORITY	1	244	V9_DOT1Q_PRIORITY	1
50	198	IN_BYTES_DELTA	8	444	V9_AS_PATH	128
1	2	V9_IN_PKTS	8	2	V9_IN_PKTS	4
2	1	V9_IN_BYTES	8	1	V9_IN_BYTES	4

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
3	10	V9_INPUT_SNMP	4	10	V9_INPUT_SNMP	4
4	14	V9_OUTPUT_SNMP	4	14	V9_OUTPUT_SNMP	4
5	22	V9_FIRST_SWITCHED	4	22	V9_FIRST_SWITCHED	4
6	21	V9_LAST_SWITCHED	4	21	V9_LAST_SWITCHED	4
7	89	V9_FORWARDING_STATUS	4	89	V9_FORWARDING_STATUS	1
8	61	V9_DIRECTION	1	61	V9_DIRECTION	1
9	302	SELECTOR_ID	4	48	V9_FLOW_SAMPLER_ID	2
10	234	V9_VRF_ID_INPUT	4	234	V9_VRF_ID_INPUT	4
11	235	V9_VRF_ID_OUTPUT	4	235	V9_VRF_ID_OUTPUT	4
12	55	V9_POST_QOS_TOS	1	55	V9_POST_QOS_TOS	1
13	8	V9_IPV4SRCADDR	4	8	V9_IPV4SRCADDR	4
14	12	V9_IPV4DSTADDR	4	12	V9_IPV4DSTADDR	4
15	7	V9_SRC_PORT	2	7	V9_SRC_PORT	2
16	11	V9_DST_PORT	2	11	V9_DST_PORT	2
17	9	V9_SRC_MASK	1	9	V9_SRC_MASK	1
18	13	V9_DST_MASK	1	13	V9_DST_MASK	1
19	4	V9_PROT	1	4	V9_PROT	1
20	6	V9_TCP_FLAGS	2	6	V9_TCP_FLAGS	1
21	5	V9_TOS	1	5	V9_TOS	1
22	52	V9_MIN_TTL	1	52	V9_MIN_TTL	1
23	53	V9_MAX_TTL	1	53	V9_MAX_TTL	1
24	54	V9_IP_IDENT	4	54	V9_IP_IDENT	4
25	197	IPFIX_FRAG_FLAGS	1	197	IPFIX_FRAG_FLAGS	1
26	88	V9_FRAGMENT_OFFSET	2	88	V9_FRAGMENT_OFFSET	2
27	184	IPFIX_TCP_SEQ_NUM	4	184	IPFIX_TCP_SEQ_NUM	4
28	25	V9_MIN_PKT_LEN	8	25	V9_MIN_PKT_LEN	8
29	26	V9_MAX_PKT_LEN	8	26	V9_MAX_PKT_LEN	8
30	503	IPFIX_L4_CHECKSUM	2	503	IPFIX_L4_CHECKSUM	2
31	504	IPFIX_ICMP_8_BYTES	8	504	IPFIX_ICMP_8_BYTES	8
32	507	GTP_TEID	4	507	GTP_TEID	4

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
33	509	GTP_QFI	1	509	GTP_QFI	1
34	510	GTP_SESS_DIR	1	510	GTP_SESS_DIR	1
35	8	V9_IPV4SRCADDR	4	8	V9_IPV4SRCADDR	4
36	12	V9_IPV4DSTADDR	4	12	V9_IPV4DSTADDR	4
37	5	V9_TOS	1	5	V9_TOS	1
38	16	V9_SRC_AS	4	16	V9_SRC_AS	4
39	17	V9_DST_AS	4	17	V9_DST_AS	4
40	18	V9_BGP_IPV4_NEXT_HOP	4	18	V9_BGP_IPV4_NEXT_HOP	4
41	63	V9_BGP_IPV6_NEXT_HOP	16	63	V9_BGP_IPV6_NEXT_HOP	16
42	15	V9_IPV4_NEXT_HOP	4	15	V9_IPV4_NEXT_HOP	4
43	62	V9_IPV6_NEXT_HOP	16	62	V9_IPV6_NEXT_HOP	16
44	56	V9_IN_SRC_MAC	6	56	V9_IN_SRC_MAC	6
45	80	V9_IN_DST_MAC	6	80	V9_IN_DST_MAC	6
46	256	V9_ETH_TYPE	2	256	V9_ETH_TYPE	2
47	243	V9_DOT1Q_VLAN_ID	2	243	V9_DOT1Q_VLAN_ID	2
48	245	V9_DOT1Q_CUST_VLAN_ID	2	245	V9_DOT1Q_CUST_VLAN_ID	2
49	244	V9_DOT1Q_PRIORITY	1	244	V9_DOT1Q_PRIORITY	1
50	198	IN_BYTES_DELTA	8	444	V9_AS_PATH	128
51				445	V9_STD_COMM	128

IPv4-GTP-IPv6 Record

This record monitors GTP-U traffic that starts in an IPv4 network and transitions into an IPv6 network, aiding in cross-network compatibility analysis.

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
1	2	V9_IN_PKTS	8	2	V9_IN_PKTS	4
2	1	V9_IN_BYTES	8	1	V9_IN_BYTES	4
3	10	V9_INPUT_SNMP	4	10	V9_INPUT_SNMP	4
4	14	V9_OUTPUT_SNMP	4	14	V9_OUTPUT_SNMP	4
5	22	V9_FIRST_SWITCHED	4	22	V9_FIRST_SWITCHED	4

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
6	21	V9_LAST_SWITCHED	4	21	V9_LAST_SWITCHED	4
7	89	V9_FORWARDING_STATUS	4	89	V9_FORWARDING_STATUS	1
8	61	V9_DIRECTION	1	61	V9_DIRECTION	1
9	302	SELECTOR_ID	4	48	V9_FLOW_SAMPLER_ID	2
10	234	V9_VRF_ID_INPUT	4	234	V9_VRF_ID_INPUT	4
11	235	V9_VRF_ID_OUTPUT	4	235	V9_VRF_ID_OUTPUT	4
12	55	V9_POST_QOS_TOS	1	55	V9_POST_QOS_TOS	1
13	27	V9_IPV6_SRC_ADDR	16	27	V9_IPV6_SRC_ADDR	16
14	28	V9_IPV6_DST_ADDR	16	28	V9_IPV6_DST_ADDR	16
15	31	V9_FLOW_LABEL	4	31	V9_FLOW_LABEL	3
16	64	V9_IPV6_OPTION_HEADERS	4	64	V9_IPV6_OPTION_HEADERS	4
17	7	V9_SRC_PORT	2	7	V9_SRC_PORT	2
18	11	V9_DST_PORT	2	11	V9_DST_PORT	2
19	30	V9_IPV6_DST_MASK	1	30	V9_IPV6_DST_MASK	1
20	29	V9_IPV6_SRC_MASK	1	29	V9_IPV6_SRC_MASK	1
21	4	V9_PROT	1	4	V9_PROT	1
22	6	V9_TCP_FLAGS	2	6	V9_TCP_FLAGS	1
23	5	V9_TOS	1	5	V9_TOS	1
24	52	V9_MIN_TTL	1	52	V9_MIN_TTL	1
25	53	V9_MAX_TTL	1	53	V9_MAX_TTL	1
26	54	V9_IP_IDENT	4	54	V9_IP_IDENT	4
27	197	IPFIX_FRAG_FLAGS	1	197	IPFIX_FRAG_FLAGS	1
28	88	V9_FRAGMENT_OFFSET	2	88	V9_FRAGMENT_OFFSET	2
29	184	IPFIX_TCP_SEQ_NUM	4	184	IPFIX_TCP_SEQ_NUM	4
30	25	V9_MIN_PKT_LEN	8	25	V9_MIN_PKT_LEN	8
31	26	V9_MAX_PKT_LEN	8	26	V9_MAX_PKT_LEN	8
32	503	IPFIX_L4_CHECKSUM	2	503	IPFIX_L4_CHECKSUM	2
33	504	IPFIX_ICMP_8_BYTES	8	504	IPFIX_ICMP_8_BYTES	8
34	507	GTP_TEID	4	507	GTP_TEID	4
35	509	GTP_QFI	1	509	GTP_QFI	1

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
36	510	GTP_SESS_DIR	1	510	GTP_SESS_DIR	1
37	8	V9_IPV4SRCADDR	4	8	V9_IPV4SRCADDR	4
38	12	V9_IPV4DSTADDR	4	12	V9_IPV4DSTADDR	4
39	5	V9_TOS	1	5	V9_TOS	1
40	16	V9_SRC_AS	4	16	V9_SRC_AS	4
41	17	V9_DST_AS	4	17	V9_DST_AS	4
42	18	V9_BGP_IPV4_NEXT_HOP	4	18	V9_BGP_IPV4_NEXT_HOP	4
43	63	V9_BGP_IPV6_NEXT_HOP	16	63	V9_BGP_IPV6_NEXT_HOP	16
44	15	V9_IPV4_NEXT_HOP	4	15	V9_IPV4_NEXT_HOP	4
45	62	V9_IPV6_NEXT_HOP	16	62	V9_IPV6_NEXT_HOP	16
46	56	V9_IN_SRC_MAC	6	56	V9_IN_SRC_MAC	6
47	80	V9_IN_DST_MAC	6	80	V9_IN_DST_MAC	6
48	256	V9_ETH_TYPE	2	256	V9_ETH_TYPE	2
49	243	V9_DOT1Q_VLAN_ID	2	243	V9_DOT1Q_VLAN_ID	2
50	245	V9_DOT1Q_CUST_VLAN_ID	2	245	V9_DOT1Q_CUST_VLAN_ID	2
51	244	V9_DOT1Q_PRIORITY	1	244	V9_DOT1Q_PRIORITY	1
52	198	IN_BYTES_DELTA	8	444	V9_AS_PATH	128
53				445	V9_STD_COMM	128

IPv6-GTP-IPv4 Record

This record monitors GTP-U traffic moving from an IPv6 network to an IPv4 network, ensuring seamless data flow across different network types.

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
1	2	V9_IN_PKTS	8	2	V9_IN_PKTS	4
2	1	V9_IN_BYTES	8	1	V9_IN_BYTES	4
3	10	V9_INPUT_SNMP	4	10	V9_INPUT_SNMP	4
4	14	V9_OUTPUT_SNMP	4	14	V9_OUTPUT_SNMP	4
5	21	V9_LAST_SWITCHED	4	21	V9_LAST_SWITCHED	4
6	22	V9_FIRST_SWITCHED	4	22	V9_FIRST_SWITCHED	4

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
7	89	V9_FORWARDING_STATUS	4	89	V9_FORWARDING_STATUS	1
8	61	V9_DIRECTION	1	61	V9_DIRECTION	1
9	302	SELECTOR_ID	4	48	V9_FLOW_SAMPLER_ID	2
10	234	V9_VRF_ID_INPUT	4	234	V9_VRF_ID_INPUT	4
11	235	V9_VRF_ID_OUTPUT	4	235	V9_VRF_ID_OUTPUT	4
12	55	V9_POST_QOS_TOS	1	55	V9_POST_QOS_TOS	1
13	8	V9_IPV4SRCADDR	4	8	V9_IPV4SRC4ADDR	4
14	12	V9_IPV4DSTADDR	4	12	V9_IPV4DSTADDR	4
15	7	V9_SRC_PORT	2	7	V9_SRC_PORT	2
16	11	V9_DST_PORT	2	11	V9_DST_PORT	2
17	9	V9_SRC_MASK	1	9	V9_SRRC_MASK	1
18	13	V9_DST_MASK	1	13	V9_DST_MASK	1
19	4	V9_PROT	1	4	V9_PROT	1
20	6	V9_TCP_FLAGS	2	6	V9_TCP_FLAGS	1
21	5	V9_TOS	1	5	V9_TOS	1
22	52	V9_MIN_TTL	1	52	V9_MIN_TTL	1
23	53	V9_MAX_TTL	1	53	V9_MAX_TTL	1
24	54	V9_IP_IDENT	4	54	V9_IP_IDENT	4
25	197	IPFIX_FRAG_FLAGS	1	197	IPFIX_FRAG_FLAGS	1
26	88	V9_FRAGMENT_OFFSET	2	88	V9_FRAGMENT_OFFSET	2
27	184	IPFIX_TCP_SEQ_NUM	4	184	IPFIX_TCP_SEQ_NUM	4
28	25	V9_MIN_PKT_LEN	8	25	V9_MIN_PKT_LEN	8
29	26	V9_MAX_PKT_LEN	8	26	V9_MAX_PKT_LEN	8
30	503	IPFIX_L4_CHECKSUM	2	503	IPFIX_L4_CHECKSUM	2
31	504	IPFIX_ICMP_8_BYTES	8	504	IPFIX_ICMP_8_BYTES	8
32	507	GTP_TEID	4	507	GTP_TEID	4
33	509	GTP_QFI	1	509	GTP_QFI	1
34	510	GTP_SESS_DIR	1	510	GTP_SESS_DIR	1
35	27	V9_IPV6_SRC_ADDR	16	27	V9_IPV6_SRC_ADDR	16
36	28	V9_IPV6_DST_ADDR	16	28	V9_IPV6_DST_ADDR	16

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
37	5	V9_TOS	1	5	V9_TOS	1
38	31	V9_FLOW_LABEL	4	31	V9_FLOW_LABEL	3
39	16	V9_SRC_AS	4	16	V9_SRC_AS	4
40	17	V9_DST_AS	4	17	V9_DST_AS	4
41	18	V9_BGP_IPV4_NEXT_HOP	4	18	V9_BGP_IPV4_NEXT_HOP	4
42	63	V9_BGP_IPV6_NEXT_HOP	16	63	V9_BGP_IPV6_NEXT_HOP	16
43	15	V9_IPV4_NEXT_HOP	4	15	V9_IPV4_NEXT_HOP	4
44	62	V9_IPV6_NEXT_HOP	16	62	V9_IPV6_NEXT_HOP	16
45	56	V9_IN_SRC_MAC	6	56	V9_IN_SRC_MAC	6
46	80	V9_IN_DST_MAC	6	80	V9_IN_DST_MAC	6
47	256	V9_ETH_TYPE	2	256	V9_ETH_TYPE	2
48	243	V9_DOT1Q_VLAN_ID	2	243	V9_DOT1Q_VLAN_ID	2
49	245	V9_DOT1Q_CUST_VLAN_ID	2	245	V9_DOT1Q_CUST_VLAN_ID	2
50	244	V9_DOT1Q_PRIORITY	1	244	V9_DOT1Q_PRIORITY	1
51	198	IN_BYTES_DELTA	8	444	V9_AS_PATH	128
52				445	V9_STD_COMM	128

IPv6-GTP-IPv6 Record

This record provides insights into GTP-U traffic within IPv6 networks, crucial for maintaining the integrity and efficiency of modern 5G infrastructures.

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
1	2	V9_IN_PKTS	8	2	V9_IN_PKTS	4
2	1	V9_IN_BYTES	8	1	V9_IN_BYTES	4
3	10	V9_INPUT_SNMP	4	10	V9_INPUT_SNMP	4
4	14	V9_OUTPUT_SNMP	4	14	V9_OUTPUT_SNMP	4
5	21	V9_LAST_SWITCHED	4	21	V9_LAST_SWITCHED	4
6	22	V9_FIRST_SWITCHED	4	22	V9_FIRST_SWITCHED	4
7	89	V9_FORWARDING_STATUS	4	89	V9_FORWARDING_STATUS	1
8	61	V9_DIRECTION	1	61	V9_DIRECTION	1

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
9	302	SELECTOR_ID	4	48	V9_FLOW_SAMPLER_ID	2
10	234	V9_VRF_ID_INPUT	4	234	V9_VRF_ID_INPUT	4
11	235	V9_VRF_ID_OUTPUT	4	235	V9_VRF_ID_OUTPUT	4
12	55	V9_POST_QOS_TOS	1	55	V9_POS_QOS_TOS	1
13	27	V9_IPV6_SRC_ADDR	16	27	V9_IPV6_SRC_ADDR	16
14	28	V9_IPV6_DST_ADDR	16	28	V9_IPV6_DST_ADDR	16
15	31	V9_FLOW_LABEL	4	31	V9_FLOW_LABEL	3
16	64	V9_IPV6_OPTION_HEADERS	4	64	V9_IPV6_OPTION_HEADERS	4
17	7	V9_SRC_PORT	2	7	V9_SRC_PORT	2
18	11	V9_DST_PORT	2	11	V9_DST_PORT	2
19	30	V9_IPV6_DST_MASK	1	30	V9_IPV6_DST_MASK	1
20	29	V9_IPV6_SRC_MASK	1	29	V9_IPV6_SRC_MASK	1
21	4	V9_PROT	1	4	V9_PROT	1
22	6	V9_TCP_FLAGS	2	6	V9_TCP_FLAGS	1
23	5	V9_TOS	1	5	V9_TOS	1
24	52	V9_MIN_TTL	1	52	V9_MIN_TTL	1
25	53	V9_MAX_TTL	1	53	V9_MAX_TTL	1
26	54	V9_IP_IDENT	4	54	V9_IP_IDENT	4
27	197	IPFIX_FRAG_FLAGS	1	197	IPFIX_FRAG_FLAGS	1
28	88	V9_FRAGMENT_OFFSET	2	88	V9_FRAGMENT_OFFSET	2
29	184	IPFIX_TCP_SEQ_NUM	4	184	IPFIX_TCP_SEQ_NUM	4
30	25	V9_MIN_PKT_LEN	8	25	V9_MIN_PKT_LEN	8
31	26	V9_MAX_PKT_LEN	8	26	V9_MAX_PKT_LEN	8
32	503	IPFIX_L4_CHECKSUM	2	503	IPFIX_L4_CHECKSUM	2
33	504	IPFIX_ICMP_8_BYTES	8	504	IPFIX_ICMP_8_BYTES	8
34	507	GTP_TEID	4	507	GTP_TEID	4
35	509	GTP_QFI	1	509	GTP_QFI	1
36	510	GTP_SESS_DIR	1	510	GTP_SESS_DIR	1
37	27	V9_IPV6_SRC_ADDR	16	27	V9_IPV6_SRC_ADDR	16
38	28	V9_IPV6_DST_ADDR	16	28	V9_IPV6_DST_ADDR	16

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
39	5	V9_TOS	1	5	V9_TOS	1
40	31	V9_FLOW_LABEL	4	31	V9_FLOW_LABEL	3
41	16	V9_SRC_AS	4	16	V9_SRC_AS	4
42	17	V9_DST_AS	4	17	V9_DST_AS	4
43	18	V9_BGP_IPV4_NEXT_HOP	4	18	V9_BGP_IPV4_NEXT_HOP	4
44	63	V9_BGP_IPV6_NEXT_HOP	16	63	V9_BGP_IPV6_NEXT_HOP	16
45	15	V9_IPV4_NEXT_HOP	4	15	V9_IPV4_NEXT_HOP	4
46	62	V9_IPV6_NEXT_HOP	16	62	V9_IPV6_NEXT_HOP	16
47	56	V9_IN_SRC_MAC	6	56	V9_IN_SRC_MAC	6
48	80	V9_IN_DST_MAC	6	80	V9_IN_DST_MAC	6
49	256	V9_ETH_TYPE	2	256	V9_ETH_TYPE	2
50	243	V9_DOT1Q_VLAN_ID	2	243	V9_DOT1Q_VLAN_ID	2
51	245	V9_DOT1Q_CUST_VLAN_ID	2	245	V9_DOT1Q_CUST_VLAN_ID	2
52	244	V9_DOT1Q_PRIORITY	1	244	V9_DOT1Q_PRIORITY	1
53	198	IN_BYTES_DELTA	8	444	V9_AS_PATH	128
54				445	V9_STD_COMM	128

Extended Template Records

IPv4 Peering Extended Record

This record extends monitoring capabilities to include detailed peering information for IPv4 traffic, enhancing traffic management and security measures.

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
1	2	V9_IN_PKTS	8	2	V9_IN_PKTS	4
2	1	V9_IN_BYTES	8	1	V9_IN_BYTES	4
3	8	V9_IPV4SRCADDR	4	8	V9_IPV4SRCADDR	4
4	12	V9_IPV4DSTADDR	4	12	V9_IPV4DSTADDR	4
5	10	V9_INPUT_SNMP	4	10	V9_INPUT_SNMP	4
6	14	V9_OUTPUT_SNMP	4	14	V9_OUTPUT_SNMP	4
7	22	V9_FIRST_SWITCHED	4	22	V9_FIRST_SWITCHED	4

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
8	21	V9_LAST_SWITCHED	4	21	V9_LAST_SWITCHED	4
9	7	V9_SRC_PORT	2	7	V9_SRC_PORT	2
10	11	V9_DST_PORT	2	11	V9_DST_PORT	2
11	16	V9_SRC_AS	4	16	V9_SRC_AS	4
12	17	V9_DST_AS	4	17	V9_DST_AS	4
13	18	V9_BGP_IPV4_NEXT_HOP	4	18	V9_BGP_IPV6_NEXT_HOP	4
14	63	V9_BGP_IPV6_NEXT_HOP	16	63	V9_BGP_IPV6_NEXT_HOP	16
15	15	V9_IPV4_NEXT_HOP	4	15	V9_IPV4_NEXT_HOP	4
16	62	V9_IPV6_NEXT_HOP	16	62	V9_IPV6_NEXT_HOP	16
17	9	V9_SRC_MASK	1	9	V9_SRC_MASK	1
18	13	V9_DST_MASK	1	13	V9_DST_MASK	1
19	4	V9_PROT	1	4	V9_PROT	1
20	6	V9_TCP_FLAGS	2	6	V9_TCP_FLAGS	1
21	5	V9_TOS	1	5	V9_TOS	1
22	55	V9_POST_QOS_TOS	1	55	V9_POST_QOS_TOS	1
23	61	V9_DIRECTION	1	61	V9_DIRECTION	1
24	89	V9_FORWARDING_STATUS	4	89	V9_FORWARDING_STATUS	1
25	302	SELECTOR_ID	4	48	V9_FLOW_SAMPLER_ID	2
26	234	V9_VRF_ID_INPUT	4	234	V9_VRF_ID_INPUT	4
27	235	V9_VRF_ID_OUTPUT	4	235	V9_VRF_ID_OUTPUT	4
28	52	V9_MIN_TTL	1	52	V9_MIN_TTL	1
29	53	V9_MAX_TTL	1	53	V9_MAX_TTL	1
30	54	V9_IP_IDENT	4	54	V9_IP_IDENT	4
31	197	IPFIX_FRAG_FLAGS	1	197	IPFIX_FRAG_FLAGS	1
32	88	V9_FRAGMENT_OFFSET	2	88	V9_FRAGMENT_OFFSET	2
33	184	IPFIX_TCP_SEQ_NUM	4	184	IPFIX_TCP_SEQ_NUM	4
34	25	V9_MIN_PKT_LEN	8	25	V9_MIN_PKT_LEN	8
35	26	V9_MAX_PKT_LEN	8	26	V9_MAX_PKT_LEN	8
36	503	IPFIX_L4_CHECKSUM	2	503	IPFIX_L4_CHECKSUM	2
37	504	IPFIX_ICMP_8_BYTES	8	504	IPFIX_ICMP_8_BYTES	8

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
38	56	V9_IN_SRC_MAC	6	56	V9_IN_SRC_MAC	6
39	80	V9_IN_DST_MAC	6	80	V9_IN_DST_MAC	6
40	256	V9_ETH_TYPE	2	256	V9_ETH_TYPE	2
41	243	V9_DOT1Q_VLAN_ID	2	243	V9_DOT1Q_VLAN_ID	2
42	245	V9_DOT1Q_CUST_VLAN_ID	2	245	V9_DOT1Q_CUST_VLAN_ID	2
43	244	V9_DOT1Q_PRIORITY	1	244	V9_DOT1Q_PRIORITY	1
44	198	IN_BYTES_DELTA	8	444	V9_AS_PATH	128
45				445	V9_STD_COMM	128

IPv6 Peering Extended Record

This record offers comprehensive peering data for IPv6 traffic, supporting advanced traffic analysis and network optimization strategies.

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
1	2	V9_IN_PKTS	8	2	V9_IN_PKTS	4
2	1	V9_IN_BYTES	8	1	V9_IN_BYTES	4
3	27	V9_IPV6_SRC_ADDR	16	27	V9_IPV6_SRC_ADDR	16
4	28	V9_IPV6_DST_ADDR	16	28	V9_IPV6_DST_ADDR	16
5	10	V9_INPUT_SNMP	4	10	V9_INPUT_SNMP	4
6	14	V9_OUTPUT_SNMP	4	14	V9_OUTPUT_SNMP	4
7	22	V9_FIRST_SWITCHED	4	22	V9_FIRST_SWITCHED	4
8	21	V9_LAST_SWITCHED	4	21	V9_LAST_SWITCHED	4
9	31	V9_FLOW_LABEL	4	31	V9_FLOW_LABEL	3
10	64	V9_IPV6_OPTION_HEADERS	4	64	V9_IPV6_OPTION_HEADERS	4
11	7	V9_SRC_PORT	2	7	V9_SRC_PORT	2
12	11	V9_DST_PORT	2	11	V9_DST_PORT	2
13	16	V9_SRC_AS	4	16	V9_SRC_AS	4
14	17	V9_DST_AS	4	17	V9_DST_AS	4
15	18	V9_BGP_IPV4_NEXT_HOP	4	18	V9_BGP_IPV6_NEXT_HOP	4
16	63	V9_BGP_IPV6_NEXT_HOP	16	63	V9_BGP_IPV6_NEXT_HOP	16

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
17	15	V9_IPV4_NEXT_HOP	4	15	V9_IPV4_NEXT_HOP	4
18	62	V9_IPV6_NEXT_HOP	16	62	V9_IPV6_NEXT_HOP	16
19	30	V9_IPV6_DST_MASK	1	30	V9_IPV6_DST_MASK	1
20	29	V9_IPV6_SRC_MASK	1	29	V9_IPV6_SRC_MASK	1
21	4	V9_PROT	1	4	V9_PROT	1
22	6	V9_TCP_FLAGS	2	6	V9_TCP_FLAGS	1
23	5	V9_TOS	1	5	V9_TOS	1
24	55	V9_POST_QOS_TOS	1	55	V9_POST_QOS_TOS	1
25	61	V9_DIRECTION	1	61	V9_DIRECTION	1
26	89	V9_FORWARDING_STATUS	4	89	V9_FORWARDING_STATUS	1
27	302	SELECTOR_ID	4	48	V9_FLOW_SAMPLER_ID	2
28	234	V9_VRF_ID_INPUT	4	234	V9_VRF_ID_INPUT	4
29	235	V9_VRF_ID_OUTPUT	4	235	V9_VRF_ID_OUTPUT	4
30	52	V9_MIN_TTL	1	52	V9_MIN_TTL	1
31	53	V9_MAX_TTL	1	53	V9_MAX_TTL	1
32	54	V9_IP_IDENT	4	54	V9_IP_IDENT	4
33	197	IPFIX_FRAG_FLAGS	1	197	IPFIX_FRAG_FLAGS	1
34	88	V9_FRAGMENT_OFFSET	2	88	V9_FRAGMENT_OFFSET	2
35	184	IPFIX_TCP_SEQ_NUM	4	184	IPFIX_TCP_SEQ_NUM	4
36	25	V9_MIN_PKT_LEN	8	25	V9_MIN_PKT_LEN	8
37	26	V9_MAX_PKT_LEN	8	26	V9_MAX_PKT_LEN	8
38	503	IPFIX_L4_CHECKSUM	2	503	IPFIX_L4_CHECKSUM	2
39	504	IPFIX_ICMP_8_BYTES	8	504	IPFIX_ICMP_8_BYTES	8
40	56	V9_IN_SRC_MAC	6	56	V9_IN_SRC_MAC	6
41	80	V9_IN_DST_MAC	6	80	V9_IN_DST_MAC	6
42	256	V9_ETH_TYPE	2	256	V9_ETH_TYPE	2
43	243	V9_DOT1Q_VLAN_ID	2	243	V9_DOT1Q_VLAN_ID	2
44	245	V9_DOT1Q_CUST_VLAN_ID	2	245	V9_DOT1Q_CUST_VLAN_ID	2
45	244	V9_DOT1Q_PRIORITY	1	244	V9_DOT1Q_PRIORITY	1
46	198	IN_BYTES_DELTA	8	444	V9_AS_PATH	128

S.No	IPFIX			NetFlow V9		
	IE #	Field	Size (Bytes)	IE #	Field	Size (Bytes)
47				445	V9_STD_COMM	128

Configure Netflow for GTP-U Traffic Monitoring

Configure a Flow Exporter

```
Router# configure
Router(config)# flow exporter-map Expo1
Router(config-fem)# source-address 2001:db8::0003
Router(config-fem)# destination 2001:db8::0002
Router(config-fem)# transport udp 1024
Router(config-fem)# version v9
Router(config-fem-ver)# options interface-table
Router(config-fem-ver)# commit
Router(config-fem-ver)# root
Router(config)#exit
```

Create a Flow Monitor for GTP-U monitoring

```
Router(config)#flow monitor-map ipv6
Router(config-fmm)#record ipv6 gtp
Router(config-fmm)#exporter Expo1
Router(config-fmm)#option bgpattr
Router(config-fmm)#cache timeout active 30
Router(config-fmm)#cache timeout inactive 5
Router(config-fmm)#exit
```

Configure a Flow Sampler

```
Router(config)# configure
Router(config)# sampler-map fsm1
Router(config-sm)# random 1 out-of 262144
Router(config)# exit
Router(config)#commit
Router(config)#exit
Router#
```

Apply a Flow Monitor Map and a Flow Sampler to a physical interface

```
Router#configure
Router(config)#interface HundredGigE 0/0/0/24
Router(config-if)#flow ipv6 monitor fmm-ipv6 sampler fsm1 ingress
Router(config-if)#commit
Router(config-if)#root
Router(config)#exit
```

Running Configuration

View the running configuration

```
Router# show run

flow exporter-map Expo1
 version v9
```

```
  options interface-table
 !
 transport udp 1024
 source-address 2001:db8::3
 destination 2001:db8::2
!
flow monitor-map fmm-ipv6
 record ipv6
 exporter Expo1
 cache entries 500000
 cache timeout active 60
 cache timeout inactive 20
!
sampler-map fsm1
 random 1 out-of 262144
!

interface HundredGigE0/0/0/24
 shutdown
 flow ipv6 monitor fmm-ipv6 sampler fsm1 ingress
!
end
```

Verification

Monitoring Cache Record for GTP-U services

In the following example, you can verify the GTP tunnel ID, QoS flow identifier, and GTP session number from the GTPTeid, GTPQFI and GTPSESSDIR field.

```
Router#show flow monitor fmm-ipv6 cache format record location 0/0/CPU0
========== Record number: 1 ==========
RecordType       : GTP Tunneled Record
IPV4SrcAddr      : 0.0.0.0
IPV4DstAddr      : 0.0.0.0
IPv6SrcAddr      : 2001:db8:1::1
IPv6DstAddr      : 2001:db8:2::2
L4SrcPort        : 0
L4DestPort       : 0
IPV4Prot         : icmpv6
IPV4TOS          : 0
InputInterface   : Gi0/2/0/0
OutputInterface  : 0
L4TCPFlags       : 0
ForwardStatus    : Fwd
FirstSwitched    : 00 00:08:59:286
LastSwitched     : 00 00:08:59:286
ByteCount        : 1296
PacketCount      : 1
Dir              : Ing
GTPTeid          : 11
GTPQFI           : 0
GTPSESSDIR       : 0
IPv6TC           : 0
IPv6FlowLabel    : 690680
MinimumTTL       : 64
MaximumTTL       : 64
IPFragFlags      : 0
IPFragOffset     : 181
IPIdentification : 0
IPV6Ident        : 1546089621
L4SequenceNum    : 0
L4Checksum       : 0
MinPktLen        : 100
```

```
MaxPktLen        : 100
ICMPBytes        : 0x8000cf945edf0002
OuterIPV4SrcAddr : 100.100.100.1
OuterIPV4DstAddr : 200.200.200.2
OuterIPv6SrcAddr : ::
OuterIPv6DstAddr : ::
BGPNextHopV4     : 0.0.0.0
BGPNextHopV6     : ::
BGPSrcOrigAS     : 0
BGPDstOrigAS     : 0
IPV4NextHop      : 0.0.0.0
IPV6NextHop      : ::
SrcMacAddr       : 00:00:3f:11:50:20
DstMacAddr       : 45:00:00:62:00:00
EthType          : 2048
Dot1qPriority    : 0
Dot1qVlanId      : 0
CustVlanId       : 0
InputVRFID       : default
OutputVRFID      : default
========== Record number: 2 ==========
RecordType       : GTP Tunneled Record
IPV4SrcAddr      : 192.168.12.2
IPV4DstAddr      : 192.168.12.1
IPv6SrcAddr      : ::
IPv6DstAddr      : ::
L4SrcPort        : 0
L4DestPort       : 0
IPV4Prot         : icmp
IPV4TOS          : 0
InputInterface   : Gi0/2/0/0
OutputInterface  : 0
L4TCPFlags       : 0
ForwardStatus    : Fwd
FirstSwitched    : 00 00:08:54:244
LastSwitched     : 00 00:08:54:244
ByteCount        : 64
PacketCount      : 1
Dir              : Ing
GTPTeid          : 11
GTPQFI           : 0
GTPSESSDIR       : 0
IPv6TC           : 0
IPv6FlowLabel    : 0
MinimumTTL       : 255
MaximumTTL       : 255
IPFragFlags      : 0
IPFragOffset     : 97
IPIdentification : 4
IPV6Ident        : 0
L4SequenceNum    : 0
L4Checksum       : 0
MinPktLen        : 100
MaxPktLen        : 100
ICMPBytes        : 0xabcdabcdabcdabcd
OuterIPV4SrcAddr : 100.100.100.1
OuterIPV4DstAddr : 200.200.200.2
OuterIPv6SrcAddr : ::
OuterIPv6DstAddr : ::
BGPNextHopV4     : 0.0.0.0
BGPNextHopV6     : ::
BGPSrcOrigAS     : 0
BGPDstOrigAS     : 0
IPV4NextHop      : 0.0.0.0
```

```
IPV6NextHop      : ::
SrcMacAddr       : 00:00:3f:11:50:20
DstMacAddr       : 45:00:00:62:00:00
EthType          : 2048
Dot1qPriority    : 0
Dot1qVlanId      : 0
CustVlanId       : 0
InputVRFID       : default
OutputVRFID      : default
```

Netflow Full Packet Capture

This feature captures the exact packet size of the ingress Netflow packet.

Earlier, when a L2VPN packet with a destination MAC address starting with the number 6 is received, the packet gets wrongly decoded as IPv6 packet; the packet size consequently gets reported inaccurately to the collector.

Configuring Netflow Full Packet Capture

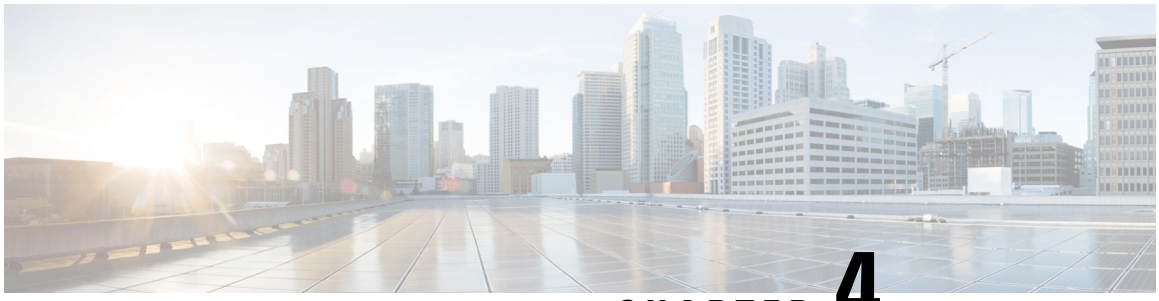
This section describes how to configure Netflow full packet capture feature on the line card location 0/1/cpu0:

Note You should reload the line card for the changes to take effect.

```
RP/0/RP0/CPU0:router(config)# hw-module profile netflow fpc-enable location 0/1/cpu0
RP/0/RP0/CPU0:router(config)# exit
RP/0/RP0/CPU0:router # system admin
RP/0/RP0/CPU0:router(sysadmin)# hw-module reload location 0/1/cpu0
RP/0/RP0/CPU0:router(sysadmin)# commit
RP/0/RP0/CPU0:router(sysadmin)# end
```

Running Config

```
config
  hw-module profile netflow fpc-enable location 0/1/cpu0
!
sysadmin
 hw-module reload location 0/1/cpu0
!
```



CHAPTER 4

IPFIX

Internet Protocol Flow Information Export (IPFIX) has been standardized by the Internet Engineering Task Force (IETF) as an export protocol for transmitting NetFlow packets. Building upon NetFlow version 9, IPFIX introduces efficient flow data formatting through templates, ensuring scalability and adaptability to diverse network environments. Utilizing UDP as the transport protocol, IPFIX facilitates the seamless transfer of NetFlow information from exporters to collectors. With native support for IPv6 flow records, the inclusion of optional data fields, and the ability to send data to multiple collectors, IPFIX proves to be a versatile and powerful solution for network administrators, enabling comprehensive traffic analysis, monitoring, and enhanced visibility into network behavior.

Restrictions

These IPFIX features are not supported:

- Variable-length information element in the IPFIX template
- Stream Control Transmission Protocol (SCTP) as the transport protocol

Limitations

- You cannot modify an exporter version of an exporter map that is already applied to an interface. To modify the exporter version, first remove the exporter configuration applied on the interface, later modify the version and apply the configuration to the interface.
- An interface can have three different monitor-maps but all the monitor maps should have the same version for the exporters. There can be different exporters for the three monitor maps but they all need to have the same exporter version either v9 or IPFIX.
- You can only have monitor-maps one of each record type attached to an interface, that is one monitor-map for IPv4 record, one monitor-map for IPv6 record and one for MPLS record. There can be different exporter maps for these three monitor-maps but all the exporter maps should have same exporter version configured, either v9 or IPFIX.
- Multiple sampler-maps can be configured but only two sampler maps can be appled to an interface across the system.

- Collect Additional BGP Information Elements for MPLS IPv4 and IPV6 Using IPFIX, on page 76
- Configuring IPFIX, on page 79
- IP Flow Information Export (IPFIX) 315, on page 87

Collect Additional BGP Information Elements for MPLS IPv4 and IPV6 Using IPFIX

Table 16: Feature History Table

Feature Name	Release Information	Feature Description
Collect Additional BGP Information Elements for MPLS IPv4 and IPV6 Using IPFIX	Release 24.1.1	You can now collect insights into how MPLS traffic is flowing through the network, assess the performance of your traffic engineering policies and make informed adjustments, pinpoint where in your MPLS network packets are being misrouted or dropped for swift troubleshooting, and also enable accurate billing for your users' customers because of insights into accurate resource usage. This is made possible because we have enabled the collection of BGP information elements for MPLS IPv4 and IPv6 traffic using IPFIX. This feature modifies the output of the **show flow monitor** command.

You can now monitor and optimize your network more effectively with IPFIX, which enhances the collection of BGP Information Elements (IEs) in IPFIX records. Specifically designed to improve congestion mitigation in core-edge link scenarios, this update introduces support for gathering eight additional BGP fields in IPFIX MPLS IPv4/IPv6 records.

Additionally, two new Information Elements, namely Minimum Time-to-Live (TTL) and Maximum TTL, are recorded. These elements provide information about the minimum Time to Live for a flow and the maximum Time to Live for a flow.

Table 17: Information Elements

IE Field	IE Number
BgpSourceAsNumber	16
BgpDestinationAsNumber	17
BgpNextHopIPv4Address	18
BgpNextHopIPv6Address	63
DestinationIPv4PrefixLength	13

IE Field	IE Number
DestinationIPv6PrefixLength	30
IpNextHopIPv4Address	15
IpNextHopIPv6Address	62
Minimum TTL	52
Maximum TTL	53

IE number, or Information Element Number, is a unique identifier assigned to specific elements within network communication protocols, facilitating standardized interpretation and management. For more information refer IP Flow Information Export (IPFIX) Entities.

Configuration

The following example shows how to collect MPLS traffic with both IPv6 and IPv4 fields.

Configuring Monitor map:

```
Router(config)#flow monitor-map mpls-1
Router(config-fmm)#record mpls ipv4-ipv6-fields
Router(config-fmm)#commit
Router(config-fmm)#exit
```

Configuring Sampler map:

```
Router(config)#sampler-map fsm1
Router(config-sm)#random 1 out-of 4000
Router(config-sm)#commit
Router(config-sm)#exit
```

Apply a Monitor Map and a Sampler Map to a physical interface

```
Router(config)#interface HundredGigE 0/0/0/24
Router(config-if)#flow mpls monitor mpls-1 sampler fsm1 ingress
Router(config-if)#exit
```

Verification

Verify the flow monitor stats statistics using the **show flow monitor cache location command**.

```
Router#show flow monitor mpls-1 cache summary location 0/0/CPU0========== Record number: 1
 ==========
========== Record number: 1 ==========
LabelType        :   Unknown
Prefix/Length    : 20.1.1.0/24
Label1-EXP-S     :    16001-0-1
Label2-EXP-S     :       -
Label3-EXP-S     :       -
Label4-EXP-S     :       -
Label5-EXP-S     :       -
Label6-EXP-S     :       -
InputInterface   : FH0/0/0/1
OutputInterface  : FH0/0/0/0
ForwardStatus    : Fwd
FirstSwitched    : 00 08:28:52:189
LastSwitched     : 00 08:28:57:649
ByteCount        : 2352
PacketCount      : 56
```

```
Dir              : Ing
SamplerID        : 1
IPV4SrcAddr      : 30.1.1.1
IPV4DstAddr      : 20.1.1.1
IPV4TOS          : 0
IPV4Prot         : udp
L4SrcPort        : 2025
L4DestPort       : 2500
L4TCPFlags       : 0
IPV4SrcPrfxLen   : 24
IPV4DstPrfxLen   : 24
BGPNextHopV4     : 192.168.10.10
BGPNextHopV6     : ::
BGPSrcOrigAS     : 2000
BGPDstOrigAS     : 1000
IPV4NextHop      : 192.168.10.10
IPV6NextHop      : ::
MinimumTTL       : 90
MaximumTTL       : 110
InputVRFID       : default
OutputVRFID      : default


========== Record number: 1 ==========
LabelType        :   Unknown
Prefix/Length    : ::/0
Label1-EXP-S     :    16001-0-1
Label2-EXP-S     :       -
Label3-EXP-S     :       -
Label4-EXP-S     :       -
Label5-EXP-S     :       -
Label6-EXP-S     :       -
InputInterface   : FH0/0/0/1
OutputInterface  : FH0/0/0/0
ForwardStatus    : Fwd
FirstSwitched    : 00 08:27:38:692
LastSwitched     : 00 08:27:47:572
ByteCount        : 5580
PacketCount      : 90
Dir              : Ing
SamplerID        : 1
IPv6SrcAddr      : 50::1
IPv6DstAddr      : 40::1
IPv6TC           : 0
IPv6FlowLabel    : 0
IPv6OptHdrs      : 0x0
IPV6Prot         : udp
L4SrcPort        : 2025
L4DestPort       : 2500
L4TCPFlags       : 0
IPV6SrcPrfxLen   : 64
IPV6DstPrfxLen   : 64
BGPNextHopV4     : 0.0.0.0
BGPNextHopV6     : ::ffff:192.168.10.10
BGPSrcOrigAS     : 2000
BGPDstOrigAS     : 1000
IPV4NextHop      : 192.168.10.10
IPV6NextHop      : ::
MinimumTTL       : 195
MaximumTTL       : 205
InputVRFID       : default
OutputVRFID      : default
```

Note When processing the ICMP Layer 4 header, the destination port is determined based on the ICMPv6 message type, instead of being set to zero. This behavior is specific to ICMPv6 and does not apply to ICMP for IPv4.

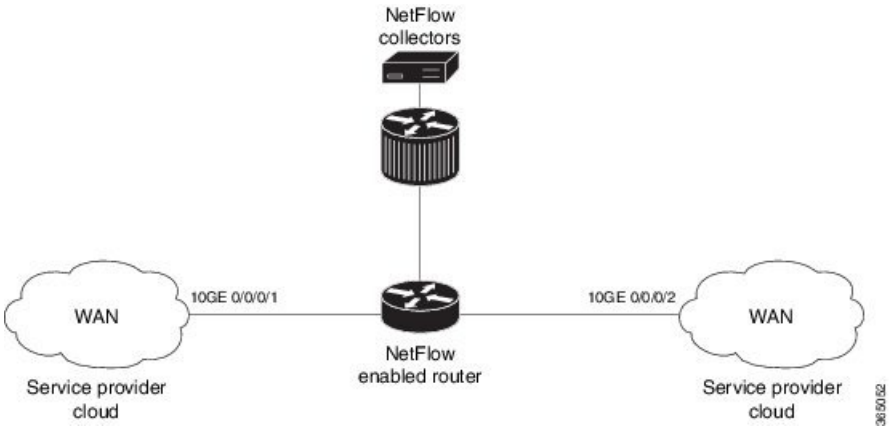
For example, in the following output, the L4DestPort value corresponds to ICMPv6 Msg Type 3 (Time Exceeded). See https://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml#icmpv6-parameters-codes-4

```
IPv6SrcAddr     : 1700::2
IPv6DstAddr     : 1800::2
BGPDstOrigAS    : 0
BGPSrcOrigAS    : 0
BGPNextHopV6    : fcbb:bb00:3::
IPv6TC          : 8
IPv6FlowLabel   : 4
IPv6OptHdrs     : 0x0
IPV6Prot        : icmpv6
MinimumTTL      : 120
MaximumTTL      : 120
L4SrcPort       : 0
L4DestPort      : 3
L4TCPFlags      : 0
IPV6DstPrfxLen  : 64
IPV6SrcPrfxLen  : 128
InputInterface  : Hu0/0/0/1
OutputInterface : Hu0/0/0/0
ForwardStatus   : Fwd
BGPNextHopV4    : 0.0.0.0
IPV4NextHop     : 0.0.0.0
IPV6NextHop     : ::
FirstSwitched   : 04 02:36:55:363
LastSwitched    : 04 02:37:19:963
ByteCount       : 25190
PacketCount     : 229
Dir             : Ing
SamplerID       : 4
SrcMacAddr      : 00:ca:ff:ee:00:01
DstMacAddr      : 04:00:00:07:1d:04
EthType         : 34525
Dot1qPriority   : 0
Dot1qVlanId     : 0
CustVlanId      : 0
InputVRFID      : vrf_1
OutputVRFID     : default
```

Configuring IPFIX

Consider SP-PE use case where SP (Service Provider) cloud is connected to the PE (Provider Edge) router through TenGigabit ethernet.

Figure 5: SP-PE Topology



Configuring NetFlow on PE router involves:

1. Configuring Exporter map with IPFIX as an exporter
2. Configuring Monitor map
3. Configuring Sampler map
4. Applying the Monitor map and Sampler map to an interface

Configuring Exporter map with IPFIX as the exporter version

```
flow exporter-map fem_ipfix
 destination 10.1.1.1
 source Loopback 0
 transport udp 1025
 exit
version ipfix
 template data timeout 600
 options sampler-table
exit
```

Configuring Monitor map

```
flow monitor-map fmm1
   record ipv4
    option filtered
   exporter fem_ipfix
   cache entries 10000
   cache timeout active 1800
   cache timeout inactive 15
   exit
```

Configuring Sampler map

```
sampler-map fsm1
 random 1 out-of 4000  /*Sampling rate supported is 1:4000*/
exit
```

Applying the Monitor map to an interface

Now apply the monitor-map **fmm1** that is configured with an exporter version IPFIX and sampler-map **fsm1** to the 10GE 0/0/0/1 interface in the ingress direction:

```
configure
 interface 10GE0/0/0/1
  flow ipv4 monitor fmm1 sampler fsm1 ingress
 exit
```

Verification

Use the **show flow flow-exporter map** command to verify the exporter version configured is IPFIX:

```
RP/0/RP0/CPU0:router# show flow exporter-map fem_ipfix
Flow Exporter Map : fem_ipfix
-------------------------------------------------
Id                  : 3
Packet-Length       : 1468
DestinationIpAddr   : 10.1.1.1
VRFName             : default
SourceIfName        : Loopback1
SourceIpAddr        : 4.4.0.1
DSCP                : 40
TransportProtocol   : UDP
TransportDestPort   : 9001

Export Version: IPFIX
  Common Template Timeout : 1800 seconds
  Options Template Timeout : 1800 seconds
  Data Template Timeout : 1800 seconds
  Interface-Table Export Timeout : 0 seconds
  Sampler-Table Export Timeout : 0 seconds
  VRF-Table Export Timeout : 0 seconds
```

Exported packets in an IPFIX packet structure are in the form of template set or data set. The first data template is sent when the configuration is activated on the interface.

With constant stream, the flowset data does not change, so data is decoded. Data template is updated in the case of timeout on the template. To change the timeout options in the flow exporter, use the `template options timeout` command:

```
RP/0/RP0/CPU0:router(config)#flow exporter-map ipfix_exp1
RP/0/RP0/CPU0:router(config-fem)#version ipfix
RP/0/RP0/CPU0:router(config-fem-ver)#template options
RP/0/RP0/CPU0:TU-PE3(config-fem-ver)#template options timeout
RP/0/RP0/CPU0:TU-PE3(config-fem-ver)#template options timeout 30

RP/0/RP0/CPU0:router# show flow exporter-map ipfix_exp1
version ipfix

  template data timeout 30
!
dscp 40
transport udp 9001
source Loopback0
destination 10.127.59.86
```

IPFIX Enablement for SRv6 and Services over SRv6 Core

Table 18: Feature History Table

Feature Name	Release Information	Description
IPFIX Enablement for SRv6 and Services over SRv6 Core	Release 7.8.1	This feature provides improved information elements about SRv6 IP traffic flows recorded by IPFIX from the network devices. The following sub-menus are introduced for this command: The **record ipv6** command is modified to support a new optional keyword, **srv6**. For more information, see: • record ipv6 • show flow monitor-map

Feature Name	Release Information	Description
Simultaneous L2 and L3 Flow Monitoring using IPFIX	Release 7.10.1	Introduced in this release on: NCS 5500 fixed port routers NCS 5500 modular routers (NCS 5500 line cards) This feature introduces support for simultaneous L2 and L3 flow monitoring. Now, you can configure IP Flow Information Export (IPFIX) to actively monitor and record end-to-end L2 and L3 flow information elements from network devices. Previously, only L2 or L3 flow could be monitored at a time. This feature introduces these changes: **CLI:** The following sub-menus are introduced for these commands: • The **record ipv4** command is modified to support a new optional keyword, **l2-l3** • The **record ipv6** command is modified to support a new optional keyword, **l2-l3** **YANG Data Model:** • New XPaths for `Cisco-IOS-XR-UM-flow-cfg.yang` (see GitHub, YANG Data Models Navigator)

When migrating from traditional IP and MPLS networks to SRv6-based networks, there is a need for information elements specific to SRv6 traffic flow. To address this, we have introduced the **srv6** keyword to the **ipv6** command. By utilizing this keyword, you can now access SRv6 flow information that is recorded by IPFIX from the network devices.

Restriction and Limitation

1. IPFIX with multiple SRH is not supported in IOS XR software version 7.10.1
2. SRv6 NetFlow is not supported on subinterfaces of decap nodes, including both L2VPN and L3VPN scenarios. To address this limitation, you can apply NetFlow on the main interface instead, which can capture traffic over the underlying subinterface and populate the record. However, please be aware that in the NetFlow record, the input ifhandle will be associated with the main interface only.

Configuration

From Cisco IOS-XR Release 7.8.1, a new optional keyword, `srv6` is introduced for the `record ipv6` option. See the following example:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config-fem)# flow monitor-map MON-MAP-v6
RP/0/RP0/CPU0:router(config-fmm)# record ipv6 srv6
RP/0/RP0/CPU0:router(config-fmm)# exporter EXP
RP/0/RP0/CPU0:router(config-fmm)# cache timeout inactive 5
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# sampler-map SAMP
RP/0/RP0/CPU0:router(config-fmm)# random 1 out-of 1000
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# interface GigabitEthernet0/1/0/0
RP/0/RP0/CPU0:router(config-fmm)# ipv6 address 2002:1::1/64
RP/0/RP0/CPU0:router(config-fmm)# flow ipv6 monitor M1 sampler SAMP ingres
```

This example shows how to display SRv6 monitor-map data for a specific flow:

```
RP/0/RP0/CPU0:router# show flow monitor-map MON

Flow Monitor Map : MON
---------------------------------------------------
Id:                1
RecordMapName:     srv6
ExportMapName:     EXP
CacheAgingMode:    Normal
CacheMaxEntries:   65535
CacheActiveTout:   101 seconds
CacheInactiveTout: 15 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

From Cisco IOS-XR Release 7.10.1, a new optional keyword, `l2-l3` is introduced for the `record ipv4` and `record ipv6` option. By utilizing this keyword, you can now access end-to-end L2 and L3 flow information that is recorded by IPFIX from the network devices. See the following example:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config-fem)# flow monitor-map M-IPv4
RP/0/RP0/CPU0:router(config-fmm)# record ipv4 l2-l3
RP/0/RP0/CPU0:router(config-fmm)# exporter EXP-ipfix
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# flow monitor-map M-IPv6
RP/0/RP0/CPU0:router(config-fmm)# record ipv6 l2-l3
RP/0/RP0/CPU0:router(config-fmm)# exporter EXP-ipfix
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# sampler-map SAMP
RP/0/RP0/CPU0:router(config-fmm)# random 1 out-of 1000
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router(config-fmm)# interface GigabitEthernet0/1/0/0
RP/0/RP0/CPU0:router(config-fmm)# description CE-PE Interface
RP/0/RP0/CPU0:router(config-fmm)# ipv4 address 1.1.1.1 255.255.255.0
RP/0/RP0/CPU0:router(config-fmm)# ipv6 address 2001:DB8:c18:1::/64
RP/0/RP0/CPU0:router(config-fmm)# flow ipv4 monitor M-IPv4 sampler SAMP ingres
```

```
RP/0/RP0/CPU0:router(config-fmm)# flow ipv6 monitor M-IPv6 sampler SAMP ingress
RP/0/RP0/CPU0:router(config-fmm)# !
RP/0/RP0/CPU0:router
```

This example shows how to display IPv4 monitor-map data for a specific flow:

```
RP/0/RP0/CPU0:router# show run flow monitor-map

flow monitor-map M-IPv4
 record ipv4 l2-l3
 exporter EXP
!
flow monitor-map M-IPv6
 record ipv6 l2-l3
 exporter EXP
!
```

This example shows how to display l2-l3 monitor-map data for IPv4 specific flow:

```
RP/0/RP0/CPU0:router# show flow monitor-map M-IPv4

Flow Monitor Map : M-IPv4
-----------------------------------------------
Id:                3
RecordMapName:     ipv4-l2-l3
ExportMapName:     EXP
CacheAgingMode:    Normal
CacheMaxEntries:   65535
CacheActiveTout:   1800 seconds
CacheInactiveTout: 15 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

This example shows how to display l2-l3 monitor-map data for IPv6 specific flow:

```
RP/0/RP0/CPU0:router# show flow monitor-map M-IPv6

Flow Monitor Map : M-IPv6
-----------------------------------------------
Id:                4
RecordMapName:     ipv6-l2-l3
ExportMapName:     EXP
CacheAgingMode:    Normal
CacheMaxEntries:   65535
CacheActiveTout:   1800 seconds
CacheInactiveTout: 15 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

This example shows the complete recorded data for SRv6 L2 services :

```
RP/0/RP0/CPU0:router# show flow monitor M-IPv6 location 0/0/CPU0

Cache summary for Flow Monitor M1:
Cache size:                         65535
Current entries:                        3
```

```
Flows added:                              4
Flows not added:                          0
Ager Polls:                           68143
  - Active timeout                        0
  - Inactive timeout                      1
  - Immediate                             0
  - TCP FIN flag                          0
  - Emergency aged                        0
  - Counter wrap aged                     0
  - Total                                 1
Periodic export:
  - Counter wrap                          0
  - TCP FIN flag                          0
Flows exported                            1

========== Record number: 1 ==========
IPv6SrcAddr           : 2::2
IPv6DstAddr           : bbbb:bc00:88:e000::
BGPDstOrigAS          : 0
BGPSrcOrigAS          : 0
BGPNextHopV6          : fe80::232:17ff:fe7e:1ce1
IPv6TC                : 0
IPv6FlowLabel         : 50686
IPv6OptHdrs           : 0x0
IPV6Prot              : 143
L4SrcPort             : 0
L4DestPort            : 0
L4TCPFlags            : 0
IPV6DstPrfxLen        : 48
IPV6SrcPrfxLen        : 128
InputInterface        : Hu0/0/0/10
OutputInterface       : BE111.1
ForwardStatus         : Fwd
FirstSwitched         : 01 18:51:25:797
LastSwitched          : 01 18:51:25:797
ByteCount             : 61004304
PacketCount           : 113814
Dir                   : Ing
SamplerID             : 1
InputVRFID            : default
OutputVRFID           : default
InnerIPV4SrcAddr      : 0.0.0.0
InnerIPV4DstAddr      : 0.0.0.0
InnerIPv6SrcAddr      : ::
InnerIPv6DstAddr      : ::
InnerL4SrcPort        : 0
InnerL4DestPort       : 0
SrcMacAddr            : 00:0c:29:0e:d8:32
DstMacAddr            : 00:0c:29:0e:d8:3c
EthType               : 2048
Dot1qPriority         : 0
Dot1qVlanId           : 2001
RecordType            : SRv6 L2 Service Record
SRHFlags              : 0x0
SRHTags               : 0x0
SRHSegmentsLeft       : 0
SRHNumSegments        : 0
```

This example shows the complete recorded data for IPv6 L2-L3 services :

```
RP/0/RP0/CPU0:router# show flow monitor M-IPv6 location 0/0/CPU0
```

```
RP/0/RP0/CPU0:router# show flow monitor MON-MAP-v6 location 0/0/CPU0
Thu Apr 28 11:36:47.622 IST
…
========== Record number: 1 ==========
IPv6SrcAddr           : 151:1::1
IPv6DstAddr           : ff02::1:ff00:2
BGPDstOrigAS          : 0
BGPSrcOrigAS          : 0
BGPNextHopV6          : ::
IPv6TC                : 224
IPv6FlowLabel         : 0
IPv6OptHdrs           : 0x0
IPV6Prot              : icmpv6
MinimumTTL            : 255
MaximumTTL            : 255
L4SrcPort             : 0
L4DestPort            : 135
L4TCPFlags            : 0
IPV6DstPrfxLen        : 0
IPV6SrcPrfxLen        : 0
InputInterface        : BE999.1
OutputInterface       : 0
ForwardStatus         : FwdNoFrag
FirstSwitched         : 01 18:51:25:797
LastSwitched          : 01 18:51:25:797
ByteCount             : 104
PacketCount           : 1
Dir                   : Ing
SamplerID             : 1
InputVRFID            : default
OutputVRFID           : default
SrcMacAddr            : 00:0c:29:0e:d8:32
DstMacAddr            : 00:0c:29:0e:d8:3c
EthType               : 2048
Dot1qPriority         : 0
Dot1qVlanId           : 100
CustVlanId            : 200
```

IP Flow Information Export (IPFIX) 315

Internet Protocol Flow Information Export (IPFIX) is an IETF standard export protocol (RFC 7011) for sending IP flow information. Cisco NCS 5500 Router supports IPFIX 315 format to export flow information. IPFIX 315 format facilitates sending ‘n’ octets frame information starting from ethernet header till transport header of the traffic flow over the network. IPFIX 315 supports sending variable size packet record with variable payload information such as IPv4, IPv6, MPLS, and Nested packets like OuterIP-GRE-InnerIP and so on. The process includes sampling and exporting the traffic flow information. Along with the ethernet frame information, IPFIX 315 format exports information of incoming and outgoing interface of the sampled packet.

Use **hw-module profile netflow ipfix315 location <** *linecard location* **>** command to enable IPFIX 315.

The information of the packets flowing through a device is used for variety of purpose including network monitoring, capacity planning, traffic management, and so on,



Note Cisco NCS 5500 Router does not support Netflow version 9 format to export flow information.

Sampling and Exporting Information

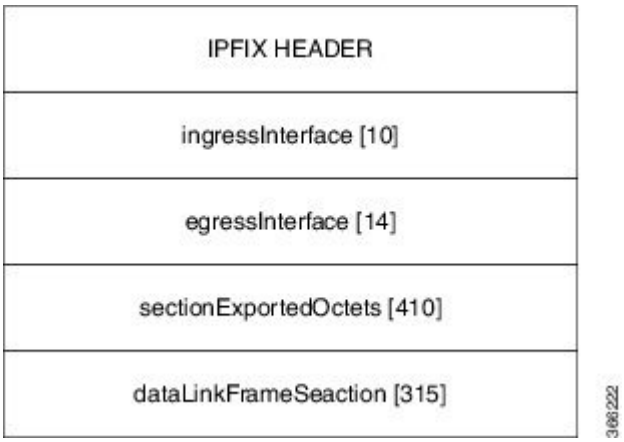
You must configure a sampling map to sample the traffic flow information. The sampler map specifies the rate at which packets (one out of n packets) are sampled. The minimun sampling rate is 1 out of 32,000 packets. Not all packets flowing through a device are exported; packets selected as per sampling rate are considered for exporting.

You must configure a sampling map to sample the traffic flow information. The sampler map specifies the rate at which packets (one out of n packets) are sampled.

- The size of the exported packet is determined by the sampled packet size from the outermost Layer 2 header.
- If the sampled packet size is less than 160 bytes, the actual packet size is exported.
- If the sampled packet size is 160 bytes or more, the exported packet size is capped at 160 bytes.
- The exported packet includes data up to the Layer 4 header if present; however, if the Layer 4 header is not found within the first 160 bytes, the export size is limited to 160 bytes.

The below figure *IPFIX 315 Export Packet Format* shows exported packet information.

Figure 6: IPFIX 315 Export Packet Format



A special cache type called Immediate Aging is used while exporting the packets. Immediate Aging ensures that the flows are exported as soon as they are added to the cache. Use the command **cache immediate** in flow monitor map configuration to enable Immediate Aging cache type.

IPFIX 315 Implementation Considerations

Here are few key points to consider before implementing IPFIX 315:

- Supported only in ingress direction.
- Supported on main interface only. The traffic on all sub-interfaces under the main interface is exported. This applies to releases up to and including IOS-XR software release 7.10.x.
- Sampling rate for bundles is per member-link and not per bundle interface.
- The outgoing interface information may not be correct incase of packets that are multicasted or broadcasted on multiple ports.

- The incoming and outgoing interface will have information of main interface and not the sub-interface even if the packet is routed via sub-interface. Incase of bundles it will point to bundle main interface.
- IPFIX 315 is not supported on BVI interface.
- Sampling and exporting of the control packets is not supported.
- When you configure **ipfix315-enable**, then you must configure all the ports on that LC with `datalinkframesection` flow.
- When the HQoS profile is enabled, Netflow does not give correct Output Interface. DSP is unique for each sub-interface.
- Netflow on the L2 interface assumes IPv4/IPv6/MPLS traffic, and if the traffic is purely L2 based, then the system ignores that traffic.
- You must remove all v9 configurations before reloading an LC. Else, with the existing v9 configurations on LC reload, you might encounter a few configuration apply error. Or, flow might be seen on an interface even when apply on interface has failed.

Configuring IPFIX 315

Configuring IPFIX 315 involves:

1. Configuring Exporter map
2. Configuring Monitor map
3. Configuring Sampler map
4. Enabling IPFIX 315 on a line card
5. Applying the Monitor map and Sampler map to an interface

Configuring Exporter map

```
flow exporter-map ipfix_exp
 version ipfix
 !
 dscp 40
 transport udp 9001
 source Loopback1
 destination 100.10.1.159
!
```

Note For **options** command and its configurations in Exporter Map, see options.

Configuring Monitor map

```
flow monitor-map ipfix_mon
 record datalinksectiondump
 exporter ipfix_exp
 cache immediate
 cache entries 1000000
```

```
 cache timeout rate-limit 1000000
!
```

Configuring Sampler map

```
sampler-map ipfix_sm
 random 1 out-of 32000
!
```



Note The default cache size is 65535, hence you can configure sampling rate as 1 out of 65535 packets. However the recommended sampling rate is 1 out of 32000 packets.

Enabling IPFIX 315 on a line card

```
(config)# hw-module profile netflow ipfix315-enable location 0/0/CPU0
```

You should reload the LC for the changes to take effect.

Applying the Monitor map to an interface

```
interface HundredGigE 0/0/0/18
        flow datalinkframesection monitor ipfix_mon sampler ipfix_sm ingress
```

Verification

Use the **show flow platform producer statistics location** command to display the IPFIX 315 ingress packets flow statistics:

```
RP/0/RP0/CPU0#show flow platform producer statistics location 0/0/CPU0
Netflow Platform Producer Counters:
IPv4 Ingress Packets:                        0
IPv4 Egress Packets:                         0
IPv6 Ingress Packets:                        0
IPv6 Egress Packets:                         0
MPLS Ingress Packets:                        0
MPLS Egress Packets:                         0
IPFIX315 Ingress Packets:               630478
IPFIX315 Egress Packets:                     0
Drops (no space):                            0
Drops (other):                               0
Unknown Ingress Packets:                     0
Unknown Egress Packets:                      0
Worker waiting:                           2443
```

Use the **show flow monitor <monitor-map> cache location** command to check the flow monitor stats. In this example flow statistics for *ipfix_mon* monitor map are displayed:

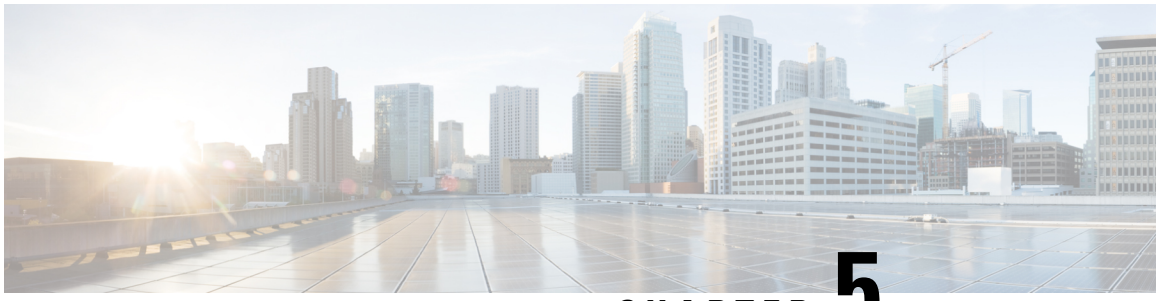
```
RP/0/RP0/CPU0#show flow monitor ipfix_mon cache location 0/0/CPU0

Cache summary for Flow Monitor ipfix_mon:
Cache size:                         65535
Current entries:                        0
Flows added:                        50399
Flows not added:                        0
Ager Polls:                          2784
  - Active timeout                      0
  - Inactive timeout                    0
  - Immediate                       50399 /*cache type immediate*/
  - TCP FIN flag                        0
```

```
  - Emergency aged                    0
  - Counter wrap aged                 0
  - Total                         50399
Periodic export:
  - Counter wrap                      0
  - TCP FIN flag                      0
Flows exported                    50399


Matching entries:                     0
```

Above example shows that there were 50399 flows added to the cache and exported.



CHAPTER 5

Configuring sFlow

This chapter describes how to configure sFlow on Cisco IOS XR devices.

- sFlow Agent, on page 93
- Guidelines and Limitations for sFlow, on page 95
- Default Settings for sFlow, on page 96
- Configuring sFlow, on page 96
- Discard Codes on sFlow, on page 100

sFlow Agent

Table 19: Feature History Table

Feature Name	Release Information	Feature Description
Sampled Flow	Release 7.5.1	Sampled flow (sFlow) allows you to monitor real-time traffic in data networks that contain switches and routers. It uses the sampling mechanism in the sFlow agent software on routers to monitor traffic and to forward the sample data to the central data collector. sFlow uses version 5 export format to forward sampled data.

Feature Name	Release Information	Feature Description
sFlow Enhancements	Release 7.3.4	With this release, the following enhancements are available with sFlow: • Discard codes 0, 1, 256, 257, and 258 are supported. This alerts the user about the packets dropped in the network. • Syslog notifications for flow monitor buffer and ring buffer status.
sFlow Enhancements	Release 7.3.3	With this release, the following enhancements are available with sFlow: • Maximum configurable sFlow datagram size allowed is greater than 1500B and up to 9KB. This allows for improved data processing by enabling the data packets to capture more network information. • Support for tunnel encapsulation, which allows for the secure movement of data from one network to the other. • Locally destined packets are now reported by sFlow output interface as format-0, value=0x3FFFFFF. This helps the users to identify the packet flows. The following new options are added to the **sflow options** command: • extended-ipv4-tunnel-egress • extended-ipv6-tunnel-egress

The sFlow Agent periodically polls the interface counters that are associated with a data source of the sampled packets. The data source can be an Ethernet interface, an EtherChannel interface, or a range of Ethernet

interfaces. The sFlow Agent queries the Ethernet port manager for the respective EtherChannel membership information and also receives notifications from the Ethernet port manager for membership changes.

When you enable sFlow sampling, based on the sampling rate and the hardware internal random number, the ingress and egress packets are sent to the CPU as an sFlow-sampled packet. The sFlow Agent processes the sampled packets and sends an sFlow datagram to the central data collector. In addition to the original sampled packet, an sFlow datagram includes the information about the ingress port, egress port, and the original packet length. An sFlow datagram can have multiple sFlow samples such as mix of flow samples and counter samples.

You can export input and ouput interface handles if the ingress or egress interface is a bundle or a BVI type. The exported interface handles are of the physical interfaces on which the packet arrived or departed and not the bundle or BVI itself.

Guidelines and Limitations for sFlow

Consider these points before configuring sFlow:

- Ingress sFlow is supported on Cisco NCS 5500 Series Routers on the line cards .
- Supports a maximum of eight export IPv4 and IPv6 destinations
- sFlow supports a maximum of two sampler maps.
- Supported sampling rate is 1 out of 262144 (maximum)
- Supports L3 Interface, L3 Bundle Interface, L3 subinterface, and L3 Bundle subinterface
- Does not support tunnel and PW-Ether interfaces.
- Supports up to 2000 L3 interfaces
- sFlow doesn't sample ARP, multicast, broadcast, and IP-in-IP packets.
- sFlow on bundle having members on different LCs have flows exported with the same ifindex id (of bundle interface, if I/O ifindex physical is not configured), but with different sub-agent id and sequence number.
- When native mode is enabled, the sFlow monitor does not export the extended router and gateway structures format check fields for MPLS traffic with IPv6 Explicit-Null labels

System Log Messages on sFlow

- The FLOW_SAMPLES_DROPPED syslog message alerts the user when the ring buffer is full. The ring buffer receives the sampled flow from the hardware process before exporting it to the sFlow collector where the analyzer tool extracts sFlow datagram.

  The following example shows the syslog message when ring buffer is full:

```
LC/0/2/CPU0:May 12 12:47:28.500 UTC: nfsvr[145]: %MGBL-FLOW-6-INFO_FLOW_SAMPLES_DROPPED
 : Flow samples for Netflow or SFlow from flow producer process got dropped in the input
 queue of the flow server process. Cumulative drop count is 98724. Possible reasons
include a high sampling rate or an increase in the traffic rate. These drops could be
transient.
```

  The ring buffer automatically returns to normal when traffic rate in the network reduces.

  The following example shows the syslog message when ring buffer returns to normal:

```
LC/0/2/CPU0:May 12 16:05:15.221 UTC: nfsvr[145]:
%MGBL-FLOW-6-INFO_FLOW_SAMPLES_DROPPING_STOPPED : Dropping of Netflow or sFlow samples
 in the input queue of flow server process has not happened for 10 minutes and operation
 is stable now. This is informational, no action required.
```

- The BUFFER_EXCEEDED syslog message alerts the user when the flow monitor buffer is full with the sampled flow.

  The following example shows the syslog message when flow monitor buffer is full:

  ```
  LC/0/11/CPU0:Dec 10 14:27:58.922 UTC: nfsvr[248]: %MGBL-SFLOW-6-INFO_BUFFER_SIZE_EXCEEDED
   : Buffer space of 256000 entries for monitor slow_mon has been exceeded. Possible
  reasons include low export rate limit and/or high sampling rate. This could be a transient
   issue too.
  ```

  The flow monitor buffer automatically returns to normal when traffic rate in the network reduces.

  The following example shows the syslog message when flow monitor buffer returns to normal:

  ```
  LC/0/11/CPU0:Dec 10 14:37:58.902 UTC: nfsvr[248]:
  %MGBL-SFLOW-6-INFO_BUFFER_EXCEEDING_STOPPED : Buffer space of 256000 entries for monitor
   slow_mon has not been exceeded for the last 10 minutes and is stable now. This is
  infomational, no action required.
  ```

Default Settings for sFlow

Here are the default sFlow parameters:

Table 20: Default Parameters for sFlow

Parameters	Default
sFlow sampling-rate	1 out of 10000 packets
sFlow sampling-size	128 bytes. The maximum configurable value for sampler size is 200 bytes.
sFlow counter-poll-interval	20 seconds
sFlow collector-port	6343

Configuring sFlow

Configuring sFlow includes:

- Configuring Exporter Map
- Configuring Monitor Map
- Configuring Sampler Map
- Configuring sFlow on an Interface
- Enabling sFlow on a Line Card

Configuring Exporter Map

This sample exporter map includes two exporter maps for IPv4 and IPv6 traffic. sFlow uses default collector-port number 6343.

Also, in the below sample configuration the DF-bit (Don't Fragment bit) is enabled for IPv4 header. However, the DF-bit configuration is not supported for IPv6 transport.



Note A DF bit is a bit within the IP header that determines whether a router is allowed to fragment a packet.

```
flow exporter-map SF-EXP-MAP-1
 version sflow v5
 !
 packet-length 1468
 transport udp 6343
 source GigabitEthernet0/0/0/1

 source-address 192.127.10.1

 destination 192.127.0.1
 dfbit set
!

flow exporter-map SF-EXP-MAP-2
 version sflow v5
 !
 packet-length 1468
 transport udp 6343
 source GigabitEthernet0/0/0/1

 source-address db8::1

destination  FF01::1
!
```

Configuring Monitor Map

This sample monitor map records sFlow traffic. Optionally, you can choose to include extended router and extended gateway information in the monitor map.

The extended router information includes:

- nexthop
- source mask length
- destination mask length

The extended gateway information includes:

- nexthop
- communities
- local preference
- AS, source AS, source peer AS, and desgtination AS path

```
flow monitor-map sflow-mon1
 record sflow
 sflow options
  input ifindex physical
  output ifindex physical
  if-counters polling-interval 10
  extended-router
  extended-gateway
 !
 exporter sflow-exp-v6-0012_99992
 cache entries 5000
 cache timeout active 5
 cache timeout inactive 10
!
```

Verification

```
show flow monitor-map sflow-mon1
Thu Nov 11 10:47:48.015 IST

Flow Monitor Map : sflow-mon1
-----------------------------------------------
Id:                6
RecordMapName:     sflow (1 labels)
ExportMapName:     sflow-exp-v4-0012_30001
                   sflow-exp-v6-0012_99992
CacheAgingMode:    Normal
CacheMaxEntries:   5000
CacheActiveTout:   5 seconds
CacheInactiveTout: 10 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50

sFlow options:
  Option: extended router
  Option: extended gateway
  Option: Input ifindex physical
  Option: Output ifindex physical
  Option: Max sample header size: using default: 128
```

Configuring Sampler Map

This sample configuration samples 1 out of 20000 packets:

Note The default sampling rate is 10000.

```
sampler-map SF-SAMP-MAP
 random 1 out-of 20000
!
```

Verification

```
Flow Exporter Map : sflow-exp-v6-0012_99992
-----------------------------------------------
Id                  : 26
```

```
Packet-Length       : 1500
DestinationIpAddr   :
VRFName             : default
SourceIfName        : Loopback0
SourceIpAddr        : ::10:0:0:3
DSCP                : 45
TransportProtocol   : UDP
TransportDestPort   : 6402
Do Not Fragment     : Enabled

Export Version: sFlow Protocol
sFlow protocol version: v5
```

Configuring sFlow on an Interface

In the following example, sFlow configuration is applied on an interface at the ingress direction:

```
interface GigabitEthernet0/0/0/3
 ipv4 address 192.127.0.56 255.255.255.0
 ipv6 address FFF2:8:DE::56/64
 ipv6 enable
 flow datalinkframesection monitor-map SF-MON-MAP sampler SF-SAMP-MAP ingress
```

Enabling sFlow on a Line Card

This sample configuration enables sFlow on a line card at node 0/0/CPU0:

```
Router(config)# hw-module profile netflow sflow-enable location 0/0/CPU0
```

You should reload the line card for the changes to take effect.

Verify sFlow Configuration

Exporter Map

To verify if the exporter map has sFlow v5 export version configured, use the **show flow monitor-map** command:

```
Router# show flow monitor-map sflow-mon1

Flow Monitor Map : sflow-mon1
-----------------------------------------------
Id:                6
RecordMapName:     sflow (1 labels)
ExportMapName:     sflow-exp-v4-0012_30001
                   sflow-exp-v6-0012_99992
CacheAgingMode:    Normal
CacheMaxEntries:   5000
CacheActiveTout:   5 seconds
CacheInactiveTout: 10 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50

sFlow options:
  Option: extended router
  Option: extended gateway
  Option: Input ifindex physical
```

```
  Option: Output ifindex physical
  Option: Max sample header size: using default: 128
```

Exporter Statistics Information

To view the flow, counter samples, and packet exported statistics, use the **show flow monitor sflow-mon1 cache location** command:

```
Router#show flow exporter SF-EXP-MAP-1 location 0/RP0/CPU0
show flow monitor sflow-mon1 cache location 0/0/cPU0
Thu Nov 11 10:57:35.168 IST
Cache summary for Flow Monitor sflow-mon1:
Cache size:                             5000
Current entries:                           0
Flows added:                          326328
Flows not added:                           0
Ager Polls:                            44656
  - Active timeout                         0
  - Inactive timeout                       0
  - Immediate                         326328
  - TCP FIN flag                           0
  - Emergency aged                         0
  - Counter wrap aged                      0
  - Total                             326328
Periodic export:
  - Counter wrap                           0
  - TCP FIN flag                           0
Flows exported                        326328
sFlow details:
  - flow samples:                     299639
  - counter samples:                   26689
     0 (0 bytes)
```

Discard Codes on sFlow

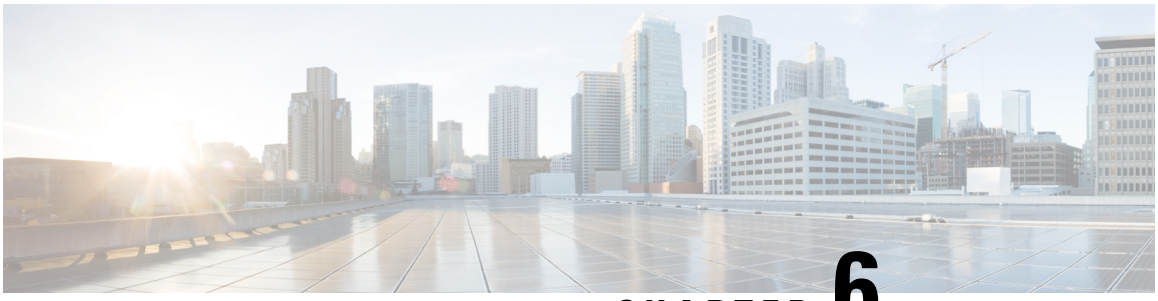
The sFlow discard code notifies the user about the packets that are dropped in the network. You can extract the discard codes from the sFlow packets using a flow-analyzer tool.

The following table lists supported discard codes on sFlow:

Table 21: Discard Codes on sFlow

Discard Code	Drop Reason	Description
0	Net unreachable	Packets dropped due to the network being unreachable
1	Host unreachable	Packets dropped due to incomplete adjacency
256	Unknown	Packets dropped due to unknown reason
257	Bad TTL	Packets dropped due to bad TTL

Discard Code	Drop Reason	Description
258	ACL Deny	Packets dropped due to denied ingress ACL

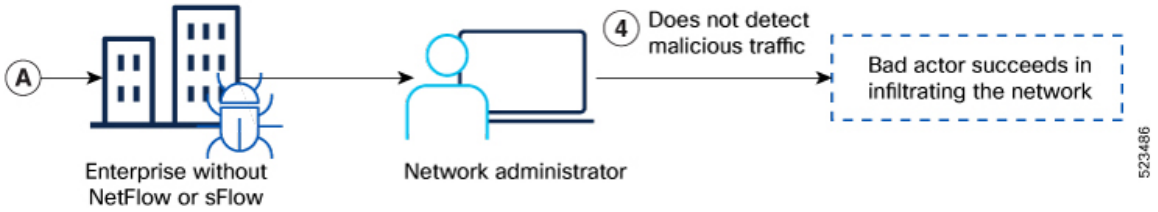


CHAPTER 6

Scenario A: Traffic Monitoring Without NetFlow and sFlow

In this particular situation, the enterprise had failed to implement any network traffic monitoring protocols such as NetFlow or sFlow.

Figure 7: Traffic monitoring without Flow data to identify malicious activity



Here is a high-level outline of the network's response to the attack:

- Bypassed threat detection and response—The network administrator does not detect any unusual network patterns or intrusions immediately following the attack.
- Successful data breach—Consequently, the network is compromised through malicious traffic that gets undetected leading to loss of critical data and trust.

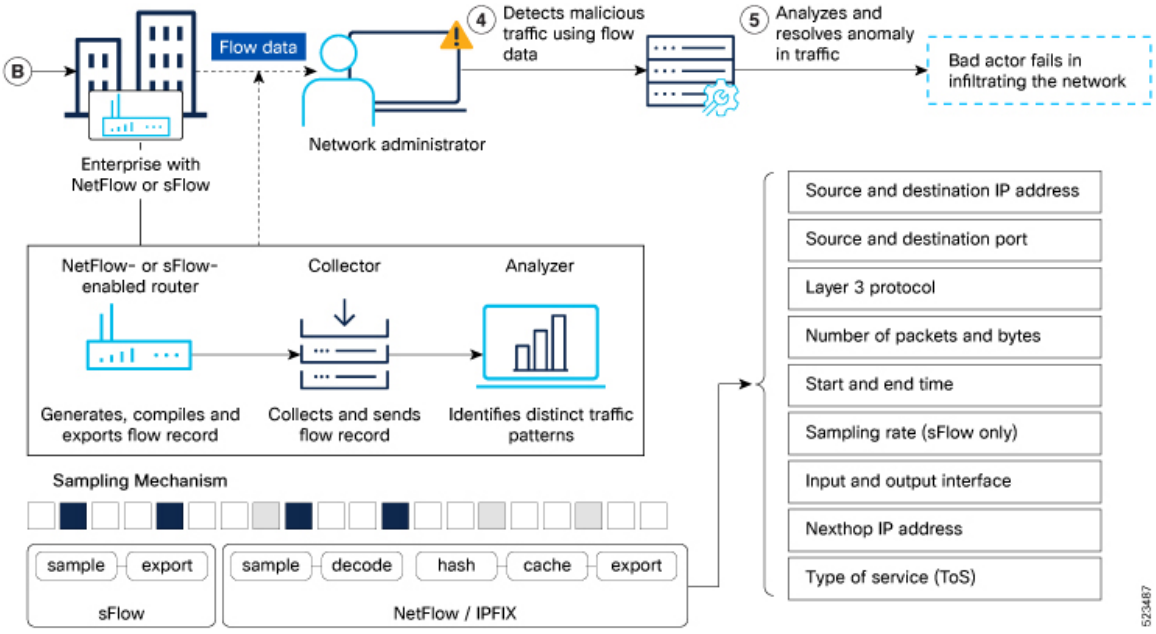
The overall network security posture is compromised due to lack of traffic monitoring mechanisms leading to poor visibility of the network and its functionalities.

- Scenario B: Traffic Monitoring With NetFlow and sFlow, on page 103

Scenario B: Traffic Monitoring With NetFlow and sFlow

In this particular situation, the enterprise has implemented network traffic monitoring protocols such as NetFlow or sFlow.

Figure 8: Traffic monitoring workflow with Flow data to identify malicious activity



Here is a high-level outline of the network's response to the attack:

- Flow data collection—Routers enabled with NetFlow or sFlow capture and retain flow records of transmitted traffic. These records store essential metadata related to the traffic's journey, including source and destination domains, the count and volume of inbound and outbound packets, timestamps and so on. The recorded flow records are then sent to a designated collector.
- Data analysis—Utilize a NetFlow or sFlow analyzer or security monitoring tool to process and analyze the collected data. The tool can identify patterns and anomalies that may indicate a security threat, such as unusual traffic patterns, unexpected communication between hosts, or a high volume of traffic from suspicious sources.
- Threat detection—The analyzer applies algorithms and rules to detect potential threats based on the analyzed data. It can compare network traffic with predefined security policies. If a potential threat is detected, the analyzer generates an alert. This alert can be sent to the network administrator for further investigation.
- Prompt investigation and responsive action—Upon receiving the alert, the network administrator can investigate the identified threat. They can analyze additional logs, inspect packet captures, or perform other security measures to gather more information about the threat. Once the threat is confirmed, appropriate actions can be taken to mitigate the impact by blocking the malicious IP addresses and isolating affected hosts to prevent further harm.

By leveraging NetFlow and sFlow for threat identification, you can proactively detect and respond to security threats, enhancing the overall network security posture. It allows for early threat detection, and faster incident response, ultimately reducing the risk of a successful attack.