Configuring NetFlow on Cisco IOS XR Software

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 or 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.

This chapter includes the following sections:

- Prerequisites for Configuring NetFlow, on page 1
- Restrictions for Configuring NetFlow, on page 1
- Information About Configuring NetFlow, on page 2
- How to Configure NetFlow on Cisco IOS XR Software, on page 16
- Additional References, on page 34

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.

- 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.

Information About Configuring NetFlow

NetFlow Overview

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

These maps are described in the sections that follow.

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

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 sampling interval of 1:1000 packets is supported on Cisco NCS 540 Series Router.

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.

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.

Table 1: 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
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.

IPFIX

Internet Protocol Flow Information Export (IPFIX) is an IETF standard export protocol for sending Netflow packets. IPFIX is based on Netflow version 9.

The IPFIX feature formats Netflow data and transfers the Netflow information from an exporter to a collector using UDP as transport protocol.

Restrictions for IPFIX

These IPFIX features are not supported:

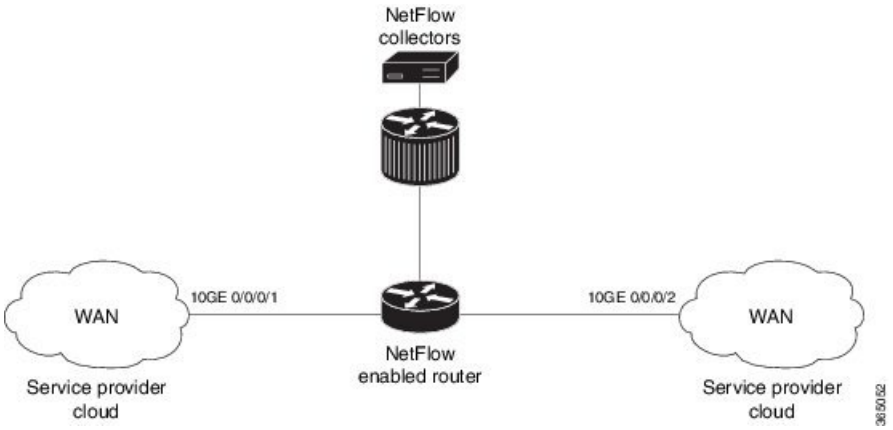
- Variable-length information element in the IPFIX template
- Stream Control Transmission Protocol (SCTP) as the transport protocol

Limitations for IPFIX

Configuring IPFIX

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

Figure 1: 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
```

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. 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.

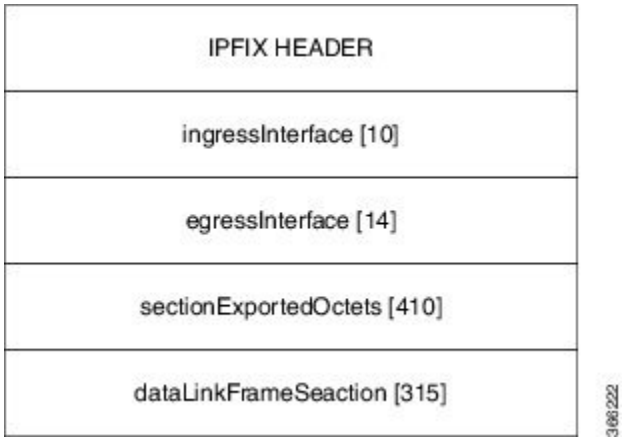
Use **hw-module profile netflow ipfix315 location** < *linecard location* > command to enable IPFIX 315.

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,

Sampling and Exporting Information

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

Figure 2: 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.
- 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. 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.

Applying the Monitor map to an interface

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

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 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
```

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
```

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 2: 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.

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
```

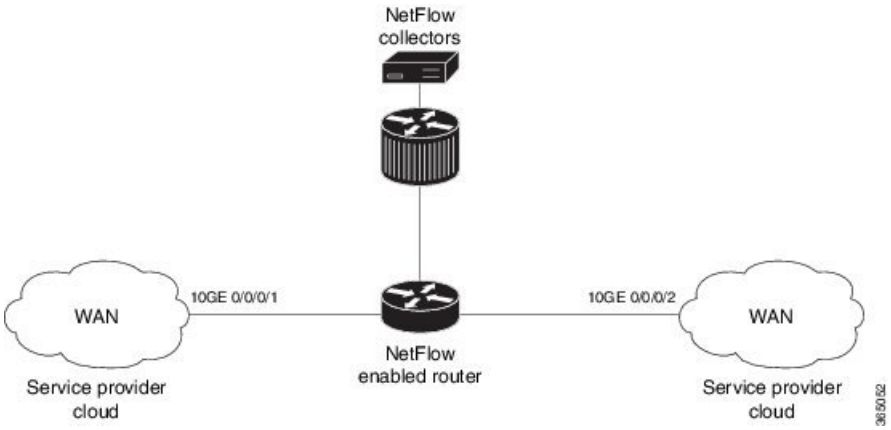
This example shows how to apply the flow monitor “MPLS-IPv6-fmm”and the sampler “FSM” to the bundle-ethernet 1 interface in the ingress direction:

```
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
```

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
```

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.

Configuring NetFlow Collection of MPLS Packets with IPv6 Fields

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 **version v9**

Example:

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

(Optional) Enters flow exporter map version configuration submode.

Step 4 **options** {**interface-table** | **sampler-table**} [**timeout** *seconds*]

Example:

```
RP/0/RP0/CPU0:router(config-fem-ver)#options interface-table timeout 300
```

(Optional) Configures the export timeout value for the interface table or the sampler table. Replace *seconds* with the export timeout value, in the range from 1 through 604800 seconds. The default is 1800 seconds for both the interface table and the sample table.

You must perform this step twice to configure the export timeout value for both an interface table and a sample table.

Step 5 **template** [**data** | **options**] **timeout** *seconds*

Example:

```
RP/0/RP0/CPU0:router(config-fem-ver)#template data timeout 300
```

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

You must perform this step twice to configure the export period for both data packets and options packets.

Step 6 **exit**

Example:

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

Exits flow exporter map version configuration mode, and enters flow exporter map configuration mode.

Step 7 **transport udp** *port*

Example:

```
RP/0/RP0/CPU0:router(config-fem)#transport udp 12515
```

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

Step 8 **source** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-fem)#source Loopback0
```

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

Step 9 **destination** *hostname_or_IP_address*

Example:

```
RP/0/RP0/CPU0:router(config-fem)#destination 170.1.1.11
```

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

Step 10 **exit**

Example:

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

Exits flow exporter map configuration mode, and enters XR Config mode.

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

Example:

```
RP/0/RP0/CPU0:router(config)#flow monitor-map MPLS-IPv6-fmm
```

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

Step 12 **record mpls** [**ipv4-ipv6-fields**] [**labels** *number*]

Example:

```
RP/0/RP0/CPU0:router(config-fmm)#record mpls ipv6-fields labels 3
```

Configures the flow record map name for IPv4, IPv6, or MPLS. Use the **ipv4-ipv6-fields** keyword to collect IPv4 and IPv6 fields in an MPLS-aware NetFlow.

Step 13 **exporter** *map_name*

Example:

```
RP/0/RP0/CPU0:router(config-fmm)#exporter exp1
```

Associates an exporter map with a monitor map.

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

Step 14 **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 15 **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 **inactive** and **active** keywords are not applicable to permanent caches.

Note The **update** keyword 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 16 **cache permanent**

Example:

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

(Optional) Disables the removal of entries from flow cache.

Step 17 **exit**

Example:

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

Exits flow monitor map configuration submode.

Step 18 **sampler-map** *map_name*

Example:

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

Creates a sampler map and enters sampler map configuration mode.

Step 19 **random 1 out-of** *sampling_interval*

Example:

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

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.

Step 20 **exit**

Example:

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

Exits sampler map configuration mode and enters XR Config mode.

Step 21 **interface** *type number*

Example:

```
RP/0/RP0/CPU0:router(config)#interface HundredGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-if)#
```

Enters interface configuration mode.

Step 22 **flow** [**ipv4** | **ipv6** | **mpls**] **monitor** *monitor_map* **sampler** *sampler_map* {**ingress**}

Example:

```
RP/0/RP0/CPU0:router(config-if)#flow ipv4 monitor MPLS-IPv6-fmm sampler fsm ingress
```

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

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 23 **commit**

Step 24 **exit**

Example:

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

Exits interface configuration submode for the Ethernet interface.

Step 25 **exit**

Example:

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

Exits XR Config mode.

Step 26 **show flow monitor-map** *map_name*

Example:

```
RP/0/RP0/CPU0: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
RP/0/RP0/CPU0:ios#
```

Displays flow monitor map data.

Step 27 **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   : 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
```

Displays exporter map data.

Example

This configuration collects MPLS traffic with IPv4 payloads.

```
RP/0/RP0/CPU0:router(config)#flow monitor-map MPLS-IPv4-fmm
RP/0/RP0/CPU0:router(config-fmm)#record mpls IPv4-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)#cache permanent
RP/0/RP0/CPU0:router(config-fmm)#exit
RP/0/RP0/CPU0:router(config)#interface HundredGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-if)#flow mpls monitor MPLS-IPv4-fmm sampler fsm ingress
```

This configuration collects MPLS traffic with IPv6 payloads.

```
RP/0/RP0/CPU0:router(config)#flow monitor-map MPLS-IPv6-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls IPv6-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)#cache permanent
RP/0/RP0/CPU0:router(config-fmm)#exit
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
```

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

```
RP/0/RP0/CPU0:router#config
RP/0/RP0/CPU0:router(config)#flow exporter-map exp1
RP/0/RP0/CPU0:router(config-fem)#version v9
RP/0/RP0/CPU0:router(config-fem-ver)#options interface-table timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)#options sampler-table timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)#template data timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)#template options timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)#exit
RP/0/RP0/CPU0:router(config-fem)#transport udp 12515
RP/0/RP0/CPU0:router(config-fem)#source Loopback0
RP/0/RP0/CPU0:router(config-fem)#destination 170.1.1.11
RP/0/RP0/CPU0:router(config-fmm)#exit
RP/0/RP0/CPU0:router(config)#flow monitor-map MPLS-IPv6-fmm
RP/0/RP0/CPU0:router(config-fmm)#record mpls ipv6-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)#exporter exp1
RP/0/RP0/CPU0:router(config-fmm)#cache entries 10000
RP/0/RP0/CPU0:router(config-fmm)#cache permanent
RP/0/RP0/CPU0:router(config-fmm)#exit

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

This configuration collects MPLS traffic with both IPv6 and IPv4 fields.

```
RP/0/RP0/CPU0:router(config)#flow monitor-map MPLS-IPv4-IPv6-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls IPv4-IPv6-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)#cache permanent
RP/0/RP0/CPU0:router(config-fmm)#exit
RP/0/RP0/CPU0:router(config)#interface HundredGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-if)#flow mpls monitor MPLS-IPv4-IPv6-fmm sampler fsm ingress
```

Note Flow records are exported using the Version 9 format.

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