

# 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:

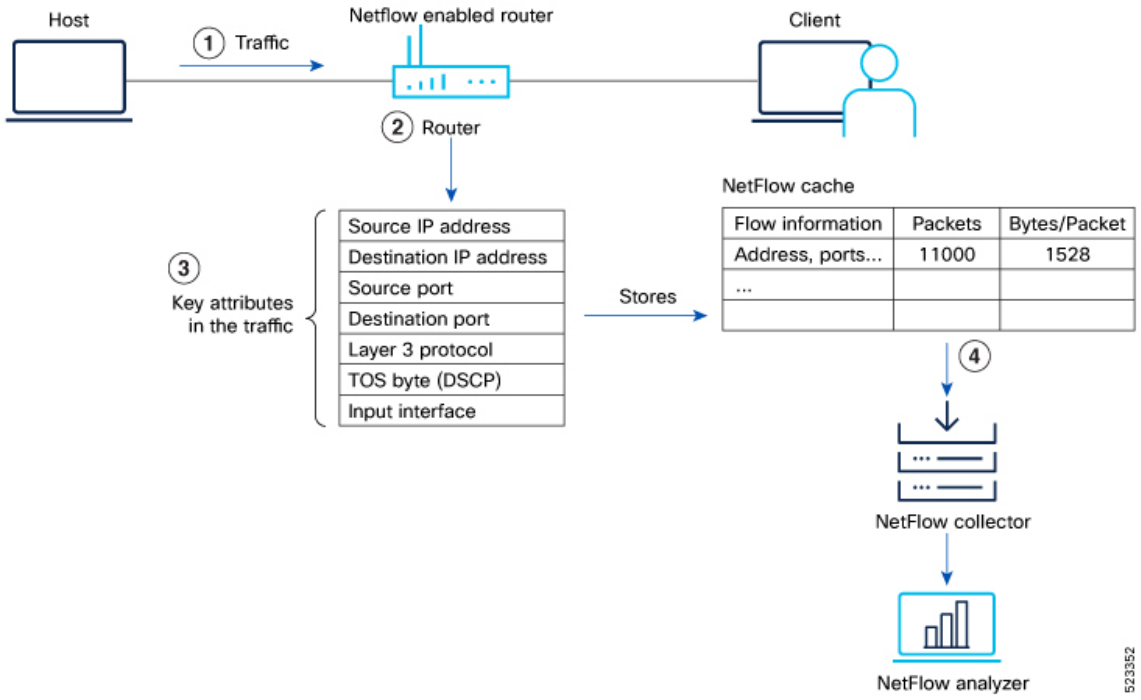
- Accounting/Billing—NetFlow data provides fine grained metering for highly flexible and detailed resource utilization accounting.
- Network Planning and Analysis—NetFlow data provides key information for strategic network planning.
- Network Monitoring—NetFlow data enables near real-time network monitoring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


- Recording of Packet Flows in NetFlow, on page 1
- Prerequisites for Configuring NetFlow, on page 3
- Restrictions for Configuring NetFlow, on page 3
- Information About Configuring NetFlow, on page 5
- How to Configure NetFlow on Cisco IOS XR Software, on page 9
- Drop Codes on NetFlow, on page 22
- Additional References, on page 22


# 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 1: 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])<br><br>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.<br><br>These IEs are added to the existing NetFlow or IPFIX template for `record ipv4` and all the IPv4 variant record types:<br>• BgpNextHopIPv6Address (IE 63)<br>• IpNextHopIPv6Address (IE 62)<br>• IpNextHopIPv4Address (IE 15)<br><br>These IEs are added to the existing NetFlow or IPFIX template for `record ipv6` and all the IPv6 variant record types:<br>• BgpNextHopIPv4Address (IE 18)<br>• IpNextHopIPv6Address (IE 62)<br>• IpNextHopIPv4Address (IE 15)<br><br>These IEs provide a detailed and structured data that is essential for various network operations and analyses.<br><br>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 2: 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.<br>Previously, the line card supported configuring Netflow sampling rate of 1:4096(4K), 1:8192(8K), and 1:16384(16K)<br>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 3: 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