NetFlow Configuration for Traffic Monitoring and Analysis

This page will help you understand the fundamental principles, variations, benefits, and limitations associated with NetFlow. Additionally, it offers guidance on configuring NetFlow.

NetFlow Essential Concepts and Terms

- **Data source:** Specific locations within the router, such as physical interfaces and VLANs, where traffic measurements can be taken.
- **Flow**: Indicates a collection of IP or MPLS packets traversing the router during a time period. All packets belonging to a particular Flow share common attributes derived from the packet's data
- **Flow record**: Is a set of key and non-key NetFlow field values used to characterize flows in the NetFlow cache. It is generated by examining packet headers, and adding a description of packet details in the NetFlow cache.
- **Exporter**: Positioned within the router that has NetFlow enabled, an Exporter monitors incoming packets, and generates Flows from them. The Exporter transmits information derived from these Flows, encapsulates as Flow Records, to the NetFlow Collector.
- **Collector**: An external device designed to receive Flow Records from one or multiple Exporters. The Collector processes the incoming export packets, and stores the associated Flow record details. Optionally, Flow records can undergo aggregation before storing it onto the hard disk.

- **NetFlow Cache**: The Cache is a segment of memory that stores flow entries prior to their exportation to an external collector. This includes two cache types: the normal cache and the permanent cache.
- **Netflow Analyser**: Is an external device or an application responsible for collecting and scrutinizing flow records to furnish valuable insights.
- **Collector address:** This comprises the IP address and a UDP port number. By default, the designated destination port number is 2055.

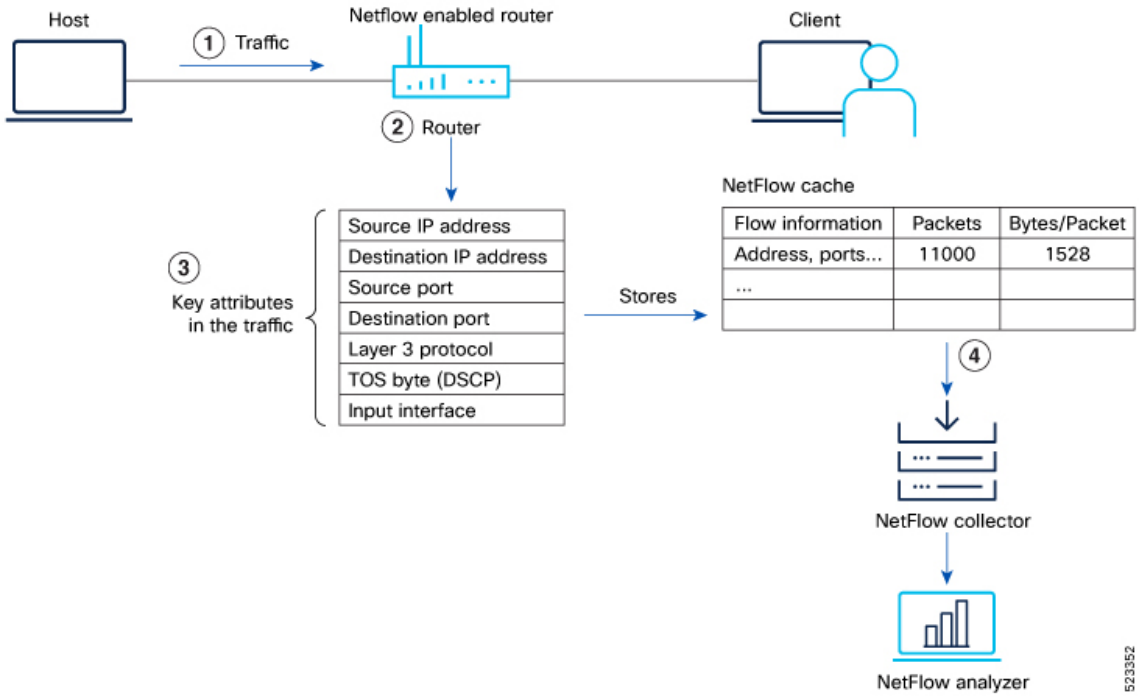
How NetFlow Works

NetFlow serves as a network monitoring protocol that facilitates the logging of metadata for each flow that traverses the router, both entering or leaving it. This protocol provides comprehensive insights into network flows, including details such as source and destination IP addresses, ports, and packet counts. It's commonly applied for traffic analysis, capacity planning, and network troubleshooting.

Recording of Packet Flows in NetFlow

The packet in NetFlow is recorded as follows:

Figure 1: Packet Flows in NetFlow



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.

Flow monitoring on Egress Interface

Egress Interface Flow Monitoring enhances network visibility and control by prioritizing outbound traffic. This capability offers advanced monitoring and management of data exiting the network, providing a more comprehensive understanding of network dynamics. The key focus of this feature is to monitor packets that are either encapsulated or decapsulated through egress sFlow.

Encapsulated and decapsulated data monitoring in sFlow serves a crucial role in safeguarding sensitive information transmitted across the network. The process involves encapsulating data with an additional layer of information, enabling verification of its authenticity and integrity. This added layer makes it challenging for attackers to intercept or modify data during transmission. Conversely, decapsulation entails removing the encapsulated data layer, empowering network devices to analyze the information and take appropriate actions

in real-time. This proactive approach aids in identifying and preventing attacks or anomalies, enhancing the overall security of the network.

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

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

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

Table 1: Information Elements

IE Field	IE Number
BgpSourceAsNumber	16
BgpDestinationAsNumber	17
BgpNextHopIPv4Address	18
BgpNextHopIPv6Address	63
DestinationIPv4PrefixLength	13
DestinationIPv6PrefixLength	30
IpNextHopIPv4Address	15
IpNextHopIPv6Address	62
Minimum TTL	52
Maximum TTL	53

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

Configuration

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

Configuring Monitor map:

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

Configuring Sampler map:

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

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

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

Verification

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

```
Router#show flow monitor mpls-1 cache summary location 0/0/CPU0========== Record number: 1
 ==========
========== Record number: 1 ==========
LabelType        :   Unknown
Prefix/Length    : 20.1.1.0/24
Label1-EXP-S     :    16001-0-1
Label2-EXP-S     :       -
Label3-EXP-S     :       -
Label4-EXP-S     :       -
Label5-EXP-S     :       -
Label6-EXP-S     :       -
InputInterface   : FH0/0/0/1
OutputInterface  : FH0/0/0/0
ForwardStatus    : Fwd
FirstSwitched    : 00 08:28:52:189
LastSwitched     : 00 08:28:57:649
ByteCount        : 2352
PacketCount      : 56
Dir              : Ing
SamplerID        : 1
IPV4SrcAddr      : 30.1.1.1
IPV4DstAddr      : 20.1.1.1
IPV4TOS          : 0
IPV4Prot         : udp
L4SrcPort        : 2025
L4DestPort       : 2500
L4TCPFlags       : 0
IPV4SrcPrfxLen   : 24
IPV4DstPrfxLen   : 24
BGPNextHopV4     : 192.168.10.10
BGPNextHopV6     : ::
BGPSrcOrigAS     : 2000
BGPDstOrigAS     : 1000
IPV4NextHop      : 192.168.10.10
IPV6NextHop      : ::
MinimumTTL       : 90
MaximumTTL       : 110
InputVRFID       : default
OutputVRFID      : default


========== Record number: 1 ==========
LabelType        :   Unknown
Prefix/Length    : ::/0
Label1-EXP-S     :    16001-0-1
Label2-EXP-S     :       -
Label3-EXP-S     :       -
Label4-EXP-S     :       -
```

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

Monitor Traffic Within Your Network

NetFlow extends its support to IPv4, IPv6, MPLS, BGP, SRv6, and GTP-U flow types, providing the capacity to monitor a diverse range of packet information.

Monitor IP Traffic

NetFlow can be used to collect traffic data for both IPv4 and IPv6 networks. The data collected includes information such as source and destination IP addresses, protocol types, port numbers, and bandwidth usage. This data can be used to identify network trends, detect security threats, and optimize network performance.

Key IP traffic attributes monitored:

- Source and Destination IP Addresses
- Source and Destination MAC Addresses
- Source and Destination Ports for TCP/User Datagram Protocol (UDP) ports
- Differentiated Services Code Point (DSCP)
- Layer 3 Protocol
- Type of Service (ToS) Byte
- Traffic receiving Interface

- Complete IPv4 Header fields, including IP-ID and TTL, among others
- Counts for Packets and Bytes
- Full Spectrum of IPv6 Header fields, encompassing Flow Label and Option Header, among others
- Flow timestamps

Monitor MPLS Traffic

NetFlow can be used to collect traffic data for MPLS traffic in a networks. NetFlow provides detailed visibility into MPLS traffic, allowing you to identify anomalies, detect cyber threats, and respond quickly to potential security incidents

Key MPLS traffic attribute monitored:

- MPLS Labels

Monitor BGP Traffic

NetFlow can be used to monitor BGP traffic in your network. NetFlow can capture BGP packets and provide information on their frequency, direction, and content. This information can be used to identify potential security threats and monitor BGP router behavior. BGP is used by network routers to exchange routing information and establish optimal paths for data to travel through a network. By monitoring BGP communication with NetFlow, you can gain valuable insights into their network's performance and ensure that their routing strategies are effective.

Key BGP traffic attributes monitored:

- Next-hop address
- Source autonomous system (AS) number
- Destination autonomous system (AS) number
- Source prefix mask
- Destination prefix mask
- BGP Next Hop
- BGP Policy Accounting traffic index

Monitor SRv6 Traffic

Table 2: Feature History Table

Feature Name	Release Information	Description
IPFIX Enablement for SRv6 and Services over SRv6 Core	Release 7.10.1	During the transition from conventional IP/MPLS networks to SRv6-based networks, the necessity for monitoring SRv6 traffic flow becomes crucial. This feature enables IPFIX to effectively monitor SRv6 IP traffic flow from network devices in the core. The feature introduces these changes: **CLI:** • The **srv6** keyword is introduced in the **record ipv6** command.

During the transition from conventional IP/MPLS networks to SRv6-based networks, the requirement for information elements specific to SRv6 traffic flow in the core arises. To address this requirement, we have introduced support for SRv6 monitoring using Netflow.

Starting from IOS-XR software release 7.10.1, you can monitor the performance of SRv6-based core networks using IPFIX.

Restriction and Limitation

- SRv6 traffic monitoring using IPFIX is supported only on P-nodes; decapsulation nodes are not supported.

Interface Types Supported with NetFlow

- Physical main interfaces
- L3 interfaces
- L3 subinterfaces
- L2 interfaces
- Bundle interfaces
- Bundle sub-interfaces
- PW-Ether interfaces

NetFlow Guidelines and Limitations

- 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 is supported.
- A valid record map name must always be configured for every flow monitor map.
- 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 field is not updated in data and flow records for the 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 NetFlow configuration is not supported.
- For Netflow IPFIX315, configure the 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.

Comparative Overview of NetFlow Version 9 and Version 10 (IPFIX)

Multiple versions of the NetFlow protocol exist. This section provides a comprehensive overview of the distinct versions within the NetFlow monitoring protocol, including NetFlow v9 and NetFlow v10 (IPFIX). It highlights the variations between these protocols.

NetFlow Version 9

NetFlow Version 9 is a template-based approach that provides flexibility in the record format. It enables enhancements to NetFlow services without concurrently altering the basic flow-record format.

NetFlow Options Template

The NetFlow Options Template serves as a distinctive template record designed to communicate the format of data associated with the NetFlow operation. Instead of sharing details about IP flows, these options serve the purpose of providing metadata pertaining to the NetFlow process itself. There are distinct options templates: the sampler options template and the interface options template. The NetFlow process exports these two tables. Furthermore, the NetFlow process also exports the VRF (Virtual Routing and Forwarding) table.

Sampler Table

The Sampler Table and Interface Option Templates play a significant role in organizing information.

The Sampler Options Template consists of a sampler table, while the Interface Option Templates consists of an interface table. Enabling these options for the sampler and interface tables simplifies the process for the collector to determine data flow information.

The sampler table offers insights into active samplers. Its primary purpose is to aid the collector in estimating the sampling rate for individual data flows. The sampler table provides the following information for each sampler:

Element ID	Field Name	Value
48	SamplerID	This ID is assigned to the sampler. It is used by the collector to retrieve information about the sampler for a data flow record.
49	SamplerMode	This field indicates the mode in which the sampling has been performed.
50	SamplerRandomInterval	This field indicates the rate at which the sampling is performed.
84	SamplerName	This field indicates the name of the sampler.

Interface Table

The interface table, contains data about interfaces that are monitored for data flow. With this data, the collector derives the interface names linked to the data flow. The interface table contains 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 mapping of VRF IDs to the VRF names. Using this information, the collector determines the name of the required VRF.

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

The VRF table consists of the following information:

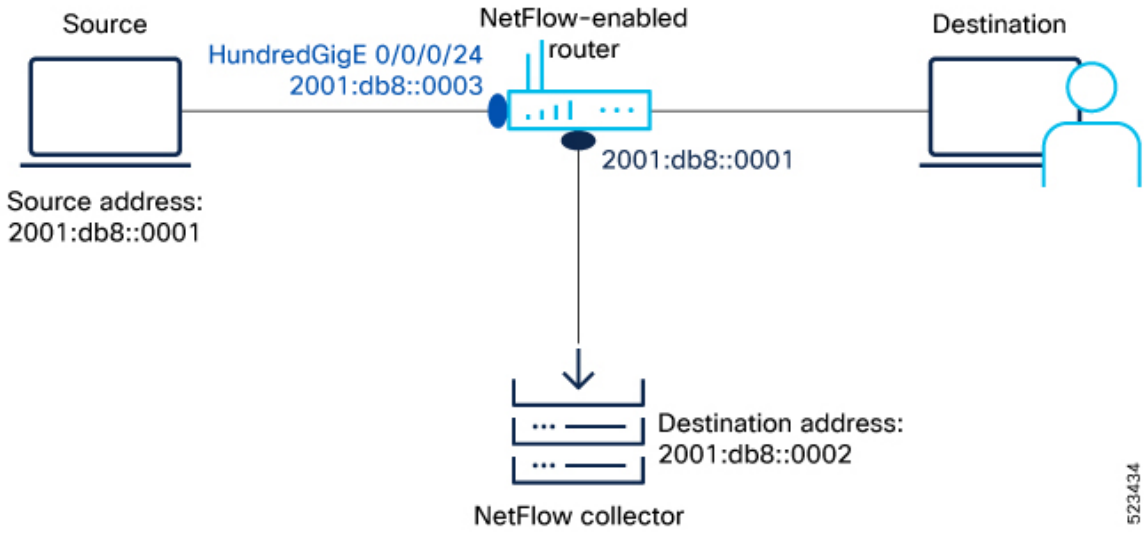
Field Name	Value
ingressVRFID	The identifier of the VRF with the name in the VRF-Name field.
VRF-Name	The VRF name has the VRFID value ingressVRFID. The value "default" indicates that the interface is not assigned explicitly to a VRF.

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 of 0 in these fields indicates that the VRF is unknown.

Configure NetFlow Version 9

Let's consider the following topology to configure NetFlow.

Figure 2: NetFlow Version 9 Configuration



To monitor traffic, you must configure one or more Flow Exporter and associate it to a Flow Monitor Flow Monitor and enable NetFlow on the interface either in egress or ingress direction. Optionally, you can configure a Flow Sampler to set the sampling rate for flow samples.

Before you begin

First, let's gather the required details to enable NetFlow on a router:

- The IP address of the source is : `2001:db8::0003`
- The IP address of the NetFlow Collector (Destination address): `2001:db8::0002`
- Interface of the router where you want to enable Netflow: `HundredGigE 0/0/0/24`
- The NetFlow version used to transport the data to the collector: `version 9`

Step 1 Configure a Flow Exporter using the flow exporter-map command to specify where and how the packets should be exported.

Example:

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

Step 2 Create a Flow Monitor using the flow monitor-map command to define the type of traffic to be monitored. You can include one or more exporter maps in the monitor map. A single flow monitor map can support up to eight exporters.

The record type specifies the type of packets that are sampled as the packets pass through the router. MPLS, IPv4, and IPv6 packet sampling is supported.

Example:

Here are the examples to record IP packets, MPLS packets, BGP packets, SRv6, and GTP-U packets.

-
```
Router#configure
Router(config)# flow monitor-map fmm-ipv6
Router(config-fmm)# record ipv6
Router(config-fmm)# cache entries 500000
Router(config-fmm)# cache timeout active 60
Router(config-fmm)# cache timeout inactive 20
Router(config-fmm)# exporter Expo1
outer(config-fmm)# commit
Router(config-fmm)# root
Router(config)#exit
```

- **MPLS Packet Monitoring:** In this example, you create a flow monitor map to record the MPLS packets.

```
Router(config)#flow monitor-map fmm-mpls-ipv6
Router(config-fmm)#record mpls ipv6-fields labels 3
Router(config-fmm)#exporter Expo1
Router(config-fmm)#cache entries 2000000
Router(config-fmm)#cache permanent
Router(config-fmm)#exit
```

- **BGP Packet Monitoring:** In this example, you create a flow monitor map to record the BGP packets with the permanent cache.

```
Router(config)# router bgp 50
Router(config-bgp)# address-family ipv6 unicast
Router(config-bgp-af)# bgp attribute-download
Router(config-bgp-af)#root
Router(config)#flow monitor-map fmm-bgp
Router(config-fmm)#record ipv6 peer-as
Router(config-fmm)#exporter Expo1
Router(config-fmm)#cache entries 2000000
Router(config-fmm)#exit
```

- **SRv6 Packet Monitoring:** Starting from Cisco IOS-XR release 7.10.1, you can create a flow monitor map to record the SRv6 packets.

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

Step 3 Configure a Flow Sampler using the sampler-map command to define the rate at which the packet sampling should be performed at the interface where NetFlow is enabled. Use the same sampler map configuration on the sub-interfaces and physical interfaces under a port.

Example:

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

Step 4 Apply a Flow Monitor Map and a Flow Sampler to a physical interface using the flow command to enable NetFlow on the router. You can choose to enable IPv4, IPv6, MPLS-aware NetFlow on the interface. Enable NetFlow in the ingress direction to monitor the incoming packets.

Note Consider these points before applying the sampler map:

- Remove any existing Netflow or sFlow configurations before applying a new Flow sampler on an interface using the no form of the command.
- Use the same sampler map configuration on the sub-interfaces and physical interfaces under a port.

Example:

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

Step 5 View the running configuration to verify the configuration that you have configured.

Example:

```
Router# show run

flow exporter-map Expo1
 version v9
  options interface-table
```

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

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

Step 6 You can verify the the above configurations using the following steps:

a) Verify the Flow Exporter configuration using the show flow exporter-map command.

Example:

```
Router#show flow exporter-map Expo1
Flow Exporter Map : Expo1
-------------------------------------------------
Id                  : 1
Packet-Length       : 1468
DestinationIpAddr   : 2001:db8::2
VRFName             : default
SourceIfName        :
SourceIpAddr        : 2001:db8::3
DSCP                : 0
TransportProtocol   : UDP
TransportDestPort   : 1024
Do Not Fragment     : Not Enabled

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

b) Verify the Flow Monitor configuration using the show flow monitor-map command.

Example:

```
Router#show flow monitor-map fmm-ipv6

Flow Monitor Map : fmm-ipv6
-------------------------------------------------
Id:                 1
RecordMapName:      ipv6
ExportMapName:      Expo1
CacheAgingMode:     Normal
CacheMaxEntries:    500000
CacheActiveTout:    60 seconds
```

```
CacheInactiveTout: 20 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

c) Verify the sampler map configuration using the show sampler-map command.

Example:

```
Router#show sampler-map fsm1

Sampler Map : fsm1
-------------------------------------------------
Id:      1
Mode:    Random (1 out of 262144 Pkts)
Router#
```

What to do next

You can now analyse the exported data using a NetFlowAnalyser.

Verify NetFlow Version 9

Verify the flows captured using the show flow monitor name cache command.

- **Cache Summary**

  In the following example, you can verify the amount of flows added and exported.

```
Router#show flow monitor fmm-ipv6 cache summary location 0/0/CPU0
Cache summary for Flow Monitor monitor1:
Cache size: 1000000
Current entries: 295
Flows added: 184409
Flows not added: 0
Ager Polls: 9824
 - Active timeout 183855
 - Inactive timeout 259
 - Immediate 0
 - TCP FIN flag 0
 - Emergency aged 0
 - Counter wrap aged 0
 - Total 184114
Periodic export:
- Counter wrap 0
- TCP FIN flag 0
Flows exported 184114
```

- **Cache Record for SRv6 L2 services**

  This example shows the complete recorded data for SRv6 L2 services

```
Router#show flow monitor fmm-ipv6 cache record location 0/0/CPU0
========== Record number: 1 ==========
IPv6SrcAddr : 2::2
IPv6DstAddr : bbbb:bc00:88:e000::
BGPDstOrigAS : 0
```

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

What to do next

You can now analyse the exported data using a NetFlowAnalyser.

Modify NetFlow Configuration

You can modify only the following flow attributes that is already applied to an interface for a monitor map, exporter map, or a sampler map.

Note that when you modify the flow attributes, the cache counters are cleared and results in resetting of the counters. As a result there could be flow accounting mismatch.

Table 3: Flow Entities and Flow Attributes that can be altered

Flow Entity	Flow Attribute	Command
Monitor map	cache timeout • active • inactive • update • rate-limit	cache timeout
	exporter	exporter
	cache entries	
	cache permanent	cache permanent
	options outphysint | bgpattr | filtered | outbundlemember	options
Exporter Map	source <*source interface*>	source
	destination <*destinaiton address*>	destination
	dscp <*dscp_value*>	dscp
	version v9 | ipfix	version ipfix version ipfix
Sampler Map	sampling interval	**sampling interval**

IPFIX (NetFlow Version 10)

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

IPFIX 315

The Internet Engineering Task Force (IETF) has standardized Internet Protocol Flow Information Export (IPFIX) as an export protocol for sending IP flow information. Router supports the IPFIX 315 format for exporting flow information. The IPFIX 315 format enables the transmission of 'n' octets of frame information starting from the Ethernet header up to the transport header of the traffic flow over the network. IPFIX 315 supports the sending of variable-sized packet records with variable payload information, such as IPv4, IPv6,

MPLS, and nested packets like OuterIP-GRE-InnerIP, and more. The process involves sampling and exporting the traffic flow information. Also, along with the Ethernet frame information, the IPFIX 315 format exports the information of the incoming and outgoing interfaces of the sampled packet.

The information of the packets flowing through a device is used for a variety of purposes, including network monitoring, capacity planning, traffic management, and more.

When exporting packets, a special cache-type called Immediate Aging is used. Immediate Aging ensures that the flows are exported as soon as they are added to the cache.

Sampling and Exporting Information

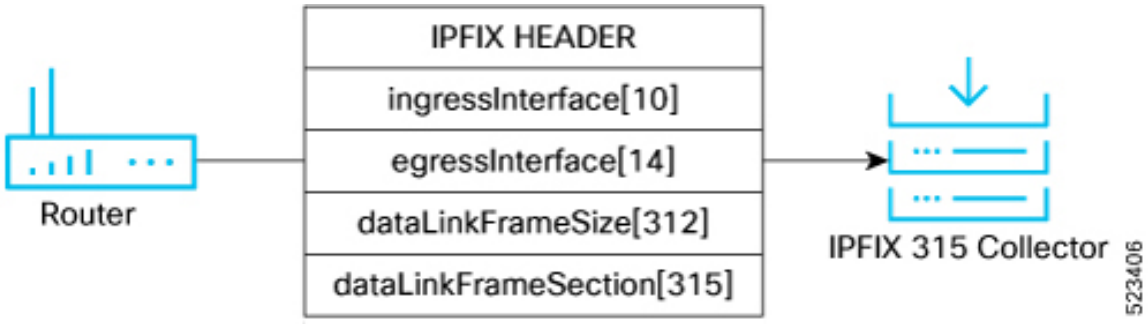
To sample the traffic flow information, configure a sampler-map that specifies the rate at which packets (one out of every 'n' packets) are sampled. Not all packets flowing through a device are exported; only the packets selected based on the sampling rate are exported.

The size of the exported packet depends on the sampled packet size and the location of the L4 header. The exported packet size is determined as follows:

- If the sampled packet size is more than 160 bytes and the L4 header is not obtained within the first 160 bytes, the exported packet size is 160 bytes.
- If the L4 header is within the first 160 bytes, the exported packet size is equal to the length of the sampled packet until the L4 header.
- If the packet size is less than 160 bytes and the L4 header isn’t within the first 160 bytes, the exported packet size is equal to the length of the packet.

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

Figure 3: IPFIX 315 Export Packet Format



Monitoring Post-QoS Data in NetFlow and IPFIX

You can now monitor information on QoS policies through the post-QoS information element field export for NetFlow and IPFIX. This improvement focuses on the analysis of packet-level information by incorporating the export of Post-QoS details, specifically the Differentiated Services Code Point (DSCP) in IPv4 and Traffic Class in IPv6. This enhancement facilitates in-depth monitoring of sampled packets, emphasizing their post-processing QoS characteristics.

In NetFlow v9 or IPFIX packets, the post-QoS field utilizes the information element postIpClassOfService (IE55) for collecting post-QoS data. Moreover, this feature brings added visibility to both pre- and post-QoS

Type of Service (TOS)/DSCP values within the show command, providing you with the data to assess network performance comprehensively.

Note The ingress QoS (i.e., pre-QoS) parameters are already exported in prior IOS-XR software releases.

- ipClassOfService - For IPv4, it represents the value of the Type of Service (TOS) field in the IPv4 packet header. For IPv6 packets, it signifies the value of the Traffic Class field in the IPv6 packet header. This information element is available prior to IOS-XR software release 24.1.1.
- postIpClassOfService - The definition of this information element is identical to the above mentioned definition of 'ipClassOfService,' except that it reports a potentially modified value caused by a router function after the packet has passed the observation point. This information element is introduced in IOS-XR software release 24.1.1.

NetFlow v9 and NetFlow v10 (IPFIX)

This section helps you understand the NetFlow v9 and NetFlow v10 (IPFIX) based on the following factors:

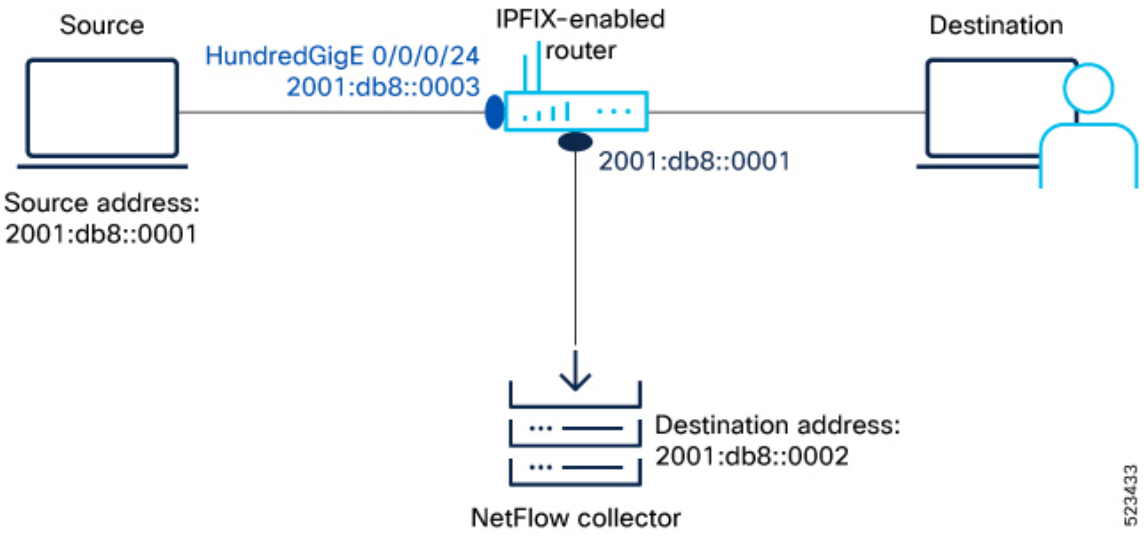
Table 4: NetFlow v9 and NetFlow v10 (IPFIX)

Factor	NetFlow v9	NetFlow v10 (IPFIX)
Transport	Typically uses UDP	Supports both UDP and TCP transport protocols
Compatibility	Compatible with older NetFlow versions	Backward-compatible with NetFlow v9
Flexibility	Fixed set of predefined fields	More flexible with variable-length information elements and custom-defined attribute
Information Elements	Limited set of predefined fields	Extensive list of predefined elements

Configure IPFIX

Let's consider the following topology to configure IPFIX:

Figure 4: NetFlow IPFIX Configuration



To monitor traffic, you must configure one or more Flow Exporter and associate it to a Flow Monitor and enable IPFIX on the interface either in egress or ingress direction. Optionally, you can configure a Flow Sampler to set the sampling rate for flow samples.

Step 1 First, let's gather the required details to enable IPFIX on a router:

- The IP address of the source : `2001:db8::0001`
- The IP address of the IPFIX Collector (Destination address): `2001:db8::0002`
- Interface of the router where we will enable IPFIX: `HundredGigE 0/0/0/24`
- NetFlow version used to transport the data to the collector: `IPFIX`

Step 2 Configure a Flow Exporter using the flow exporter-map command to specify where and how the packets should be exported.

```
Router(config)# flow exporter-map fem_ipfix
Router(config-fem)# destination 2001:db8::0002
Router(config-fem)# source Loopback 0
Router(config-fem)# transport udp 9001
Router(config-fem)# exit
Router(config-fem)# version ipfix
Router(config-fem-ipfix)# template data timeout 600
Router(config-fem-ipfix)# options interface-table
Router(config-fem-ipfix)# exit
```

Verify the Flow Exporter configuration using the show flow exporter-map command.

```
Router#show exporter-map fem_ipfix
Flow Exporter Map : fem_ipfix
-------------------------------------------------
Id                    : 1
Packet-Length         : 1468
DestinationIpAddr     : 2001:db8::2
VRFName               : default
```

```
SourceIfName         :
SourceIpAddr         : 2001:db8::3
DSCP                 : 0
TransportProtocol    : UDP
TransportDestPort    : 1024
Do Not Fragment      : Not Enabled

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

Step 3 Create a Flow Monitor using the flow monitor-map command to define the type of traffic to be monitored. You can include one or more exporter maps in the monitor map. A single flow monitor map can support up to eight exporters.

The record type specifies the type of packets that are sampled as the packets pass through the router. MPLS, IPv4, and IPv6 packet sampling is supported.

```
Router(config)# flow monitor-map fmm1
Router(config-fmm)# record ipv6
Router(config-fmm)# option filtered
Router(config-fmm)# exporter fem_ipfix
Router(config-fmm)# cache entries 65535
Router(config-fmm)# cache timeout active 1800
Router(config-fmm)# cache timeout inactive 15
Router(config-fmm)# exit
```

Verify the Flow Monitor configuration using the show flow monitor-map command.

```
Router#show flow monitor-map fmm1

Flow Monitor Map : fmm1
-----------------------------------------------
Id:                1
RecordMapName:     ipv6
ExportMapName:     Expo1
CacheAgingMode:    Normal
CacheMaxEntries:   500000
CacheActiveTout:   60 seconds
CacheInactiveTout: 20 seconds
CacheUpdateTout:   N/A
CacheRateLimit:    2000
HwCacheExists:     False
HwCacheInactTout:  50
```

Step 4 Configure a Flow Sampler using the sampler-map command. .Use the same sampler map configuration on the sub-interfaces and physical interfaces under a port.

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

Verify the sampler map configuration using the show sampler-map command.

```
Router#show sampler-map fsm1

Sampler Map : fsm1
```

```
------------------------------------------------
Id:       1
Mode:     Random (1 out of 4000 Pkts)
Router#
```

Step 5 View the running configuration to verify the configuration that you have configured.

Step 6 Apply a Monitor Map and a Sampler Map to a physical interface using the command to enable IPFIX on the router.

```
Router(config)#
Router(config-if)#flow ipv4 monitor fmm1 sampler fsm1 ingress
Router(config-if)#exit
```

Configure IPFIX 315

This section provides you instructions to enable IPFIX 315 on Cisco IOS XR Software.

Step 1 Enable IPFIX 315 for flow monitoring.

```
Router(config)# hw-module profile netflow ipfix315-enable
```

Step 2 Configure an exporter map with IPFIX as the exporter version using the flow exporter-map command in global configuration mode to specify where and how the packets should be exported.

```
Router(config)# flow exporter-map ipfix_exp
Router(config-fem)# version ipfix
Router(config-fem-ipfix)# template data timeout 10
Router(config-fem)# dscp 63
Router(config-fem)# transport udp 12000
Router(config-fem)# source Loopback 0
Router(config-fem)# destination 100.10.1.159
Router(config-fem)# exit
```

Step 3 Create a flow monitor using the flow monitor-mapcommand in global configuration mode to define the type of traffic to be monitored. You can include one or more exporter maps in the monitor map.

```
Router(config)# flow monitor-map ipfix_mon
Router(config-fmm)# record datalinksectiondump
Router(config-fmm)# exporter ipfix_exp
Router(config-fmm)# cache immediate
Router(config-fmm)# exit
```

Step 4 Configure a sampler map using the sampler-mapcommand to define the rate at which the packet sampling should be performed at the interface where IPFIX is enabled.

```
Router# sampler-map ipfix_sm
Router(config-sm)# random 1 out-of 32000
Router(config)# exit
```

Step 5 Apply a monitor map and a Sampler Map to a physical interface using the flowcommand to enable IPFIX on the router.

```
Router(config)#
Router(config-if)#ipv4 address 192.1.108.2 255.255.255.0
Router(config-if)#ipv6 address 1:108::2/64
```

```
Router(config-if)#flow datalinkframesection monitor ipfix_mon sampler ipfix_sm ingress
Router(config-if)#encapsulation dot1q 139
```

Step 6 Verify the sampled and exported flow statistics using the show flow platform producer statistics location command.

In this show output, you can see that the system has actively received and monitored a total of 630,478 IPFIX 315 packets.

```
Router#
Netflow Platform Producer Counters:
IPv4 Ingress Packets: 0
IPv4 Egress Packets: 0
IPv6 Ingress Packets: 0
IPv6 Egress Packets: 0
MPLS Ingress Packets: 0
MPLS Egress Packets: 0
IPFIX315 Ingress Packets: 630478
IPFIX315 Egress Packets: 0
Drops (no space): 0
Drops (other): 0
Unknown Ingress Packets: 0
Unknown Egress Packets: 0
Worker waiting: 2443
```

Step 7 Verify the flow monitor stats statistics using the show flow monitor cache location command.

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

```
Router#
Cache summary for Flow Monitor ipfix_mon:
Cache size: 65535
Current entries: 0
Flows added: 50399
Flows not added: 0
Ager Polls: 2784
- Active timeout 0
- Inactive timeout 0
- Immediate 50399
- TCP FIN flag 0
- Emergency aged 0
- Counter wrap aged 0
- Total 50399
Periodic export:
- Counter wrap 0
- TCP FIN flag 0
Flows exported 50399
Matching entries: 0
```