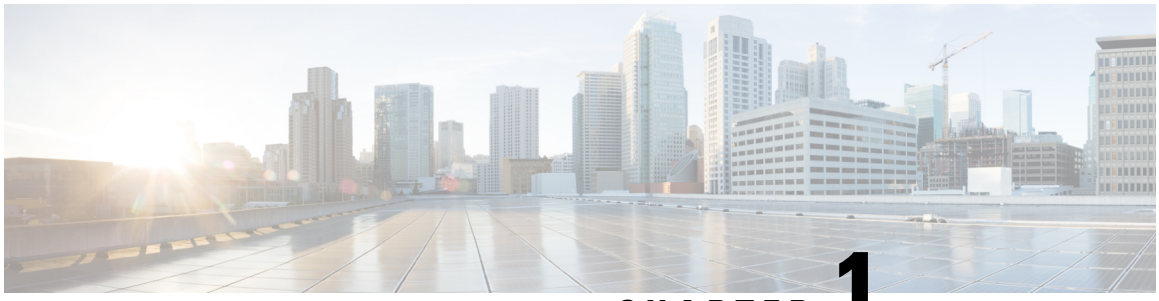
Flexible Netflow Configuration Guide, Cisco IOS Release 15M&T

First Published: 2012-11-26

Last Modified: 2012-11-26

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

CONTENTS



CHAPTER 1

Flexible Netflow Overview

NetFlow is a Cisco IOS technology that provides statistics on packets flowing through the router. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides data to enable network and security monitoring, network planning, traffic analysis, and IP accounting.

Flexible NetFlow improves on original NetFlow by adding the capability to customize the traffic analysis parameters for your specific requirements. Flexible NetFlow facilitates the creation of more complex configurations for traffic analysis and data export through the use of reusable configuration components.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Prerequisites for Flexible NetFlow

- You are familiar with the Flexible NetFlow key fields as they are defined in the following commands:
  - **match flow**
  - **match interface**
  - **match** {**ipv4** | **ipv6**}
  - **match routing**

- **match transport**

- You are familiar with the Flexible NetFlow nonkey fields as they are defined in the following commands:
  - **collect counter**
  - **collect flow**
  - **collect interface**
  - **collect {ipv4 | ipv6}**
  - **collect routing**
  - **collect timestamp sys-uptime**
  - **collect transport**
- The networking device must be running a Cisco release that supports Flexible NetFlow.

IPv4 Traffic

- The networking device must be configured for IPv4 routing.
- One of the following must be enabled on your device and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding or distributed Cisco Express Forwarding.

IPv6 Traffic

- The networking device must be configured for IPv6 routing.
- One of the following must be enabled on your device and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding IPv6 or distributed Cisco Express Forwarding.

Restrictions for Flexible Netflow

- Flexible NetFlow is not supported on Switch Virtual Interface (SVI).
- It is recommended that the total dataplane memory consumed by Flexible Netflow or Original Netflow is limited to a maximum of 25% of the amount of data plane DRAM for an ESP/FP.
- Flexible Netflow export will not work over an IPSEC VPN tunnel if the source of the netflow data is the same router where the VPN tunnel is terminated unless you configure the output-features command under the flow exporter.
- Flexible NetFlow does not monitor PPPoE traffic flowing through a Catalyst 6500 Series switch with Supervisor Engine 2T.

Information About Flexible Netflow

Flexible NetFlow Overview

Flexible NetFlow facilitates the creation of more complex configurations for traffic analysis and data export through the use of reusable configuration components.

Typical Uses for NetFlow

NetFlow is typically used for several key customer applications, including the following:

- Network monitoring. NetFlow data enables extensive near-real-time network monitoring capabilities. Flow-based analysis techniques are used by network operators to visualize traffic patterns associated with individual routers and switches and network-wide traffic patterns (providing aggregate traffic or application-based views) to provide proactive problem detection, efficient troubleshooting, and rapid problem resolution.
- Application monitoring and profiling. NetFlow data enables network managers to gain a detailed time-based view of application usage over the network. This information is used to plan, understand new services, and allocate network and application resources (for example, web server sizing and VoIP deployment) to meet customer demands responsively.
- User monitoring and profiling. NetFlow data enables network engineers to gain detailed understanding of customer and user use of network and application resources. This information may then be used to efficiently plan and allocate access, backbone, and application resources and to detect and resolve potential security and policy violations.
- Network planning. NetFlow can be used to capture data over a long period of time, affording the opportunity to track and anticipate network growth and plan upgrades to increase the number of routing devices, ports, and higher-bandwidth interfaces. NetFlow services data optimizes network planning for peering, backbone upgrades, and routing policy. NetFlow helps to minimize the total cost of network operations while maximizing network performance, capacity, and reliability. NetFlow detects unwanted WAN traffic, validates bandwidth and quality of service (QoS), and allows the analysis of new network applications. NetFlow will give you valuable information to reduce the cost of operating your network.
- Security analysis. NetFlow identifies and classifies distributed denial of service (dDoS) attacks, viruses, and worms in real time. Changes in network behavior indicate anomalies that are clearly demonstrated in Flexible NetFlow data. The data is also a valuable forensic tool to understand and replay the history of security incidents.
- Billing and accounting. NetFlow data provides fine-grained metering (for instance, flow data includes details such as IP addresses, packet and byte counts, time stamps, type of service (ToS), and application ports) for highly flexible and detailed resource utilization accounting. Service providers may use the information for billing based on time of day, bandwidth usage, application usage, quality of service, and so on. Enterprise customers may use the information for departmental charge back or cost allocation for resource utilization.
- NetFlow data warehousing and data mining. NetFlow data (or derived information) can be warehoused for later retrieval and analysis in support of proactive marketing and customer service programs (for example, discovering which applications and services are being used by internal and external users and targeting them for improved service, advertising, and so on). In addition, Flexible NetFlow data gives market researchers access to the "who," "what," "where," and "how long" information relevant to enterprises and service providers.

Use of Flows in Original NetFlow and Flexible NetFlow

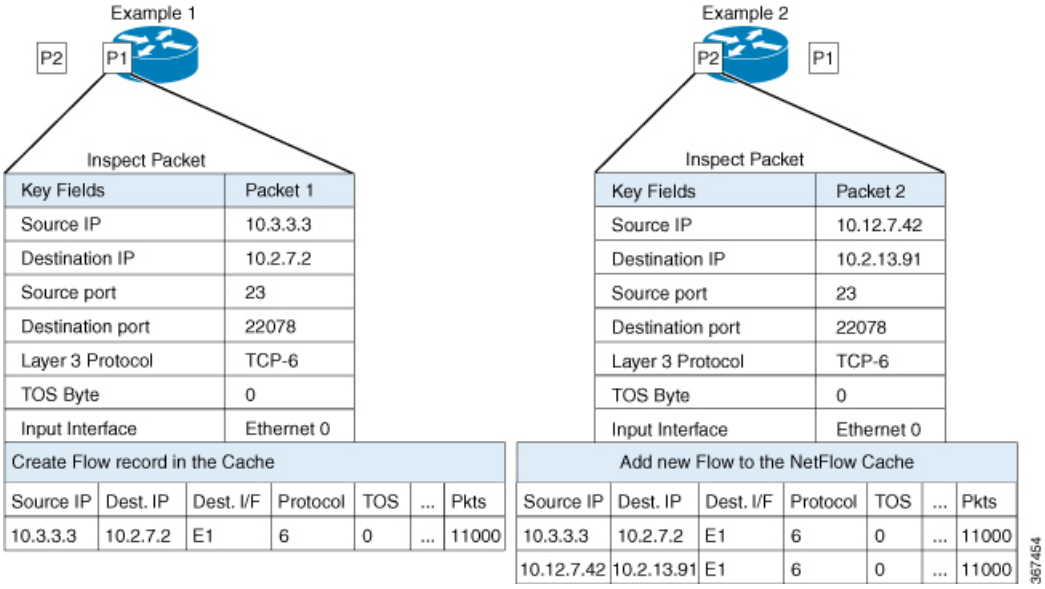
Original NetFlow and Flexible NetFlow both use the concept of flows. A *flow* is defined as a stream of packets between a given source and a given destination.

Original NetFlow and Flexible NetFlow both use the values in key fields in IP datagrams, such as the IP source or destination address and the source or destination transport protocol port, as the criteria for determining when a new flow must be created in the cache while network traffic is being monitored. When the value of the data in the key field of a datagram is unique with respect to the flows that already exist, a new flow is created.

Original NetFlow and Flexible NetFlow both use nonkey fields as the criteria for identifying fields from which data is captured from the flows. The flows are populated with data that is captured from the values in the nonkey fields.

The figure below is an example of the process for inspecting packets and creating flow records in the cache. In this example, two unique flows are created in the cache because different values are in the source and destination IP address key fields.

Figure 1: Packet Inspection



Original NetFlow and Benefits of Flexible NetFlow

Original NetFlow uses a fixed seven tuples of IP information to identify a flow.

Flexible NetFlow allows the flow to be user defined. The benefits of Flexible NetFlow include:

- High-capacity flow recognition, including scalability and aggregation of flow information.
- Enhanced flow infrastructure for security monitoring and dDoS detection and identification.
- New information from packets to adapt flow information to a particular service or operation in the network. The flow information available will be customizable by Flexible NetFlow users.
- Extensive use of Cisco's flexible and extensible NetFlow Version 9 and version 10 export formats. With version 10 export format, support for variable length field for the wireless client's SSID is available.
- A comprehensive IP accounting feature that can be used to replace many accounting features, such as IP accounting, Border Gateway Protocol (BGP) Policy Accounting, and persistent caches.

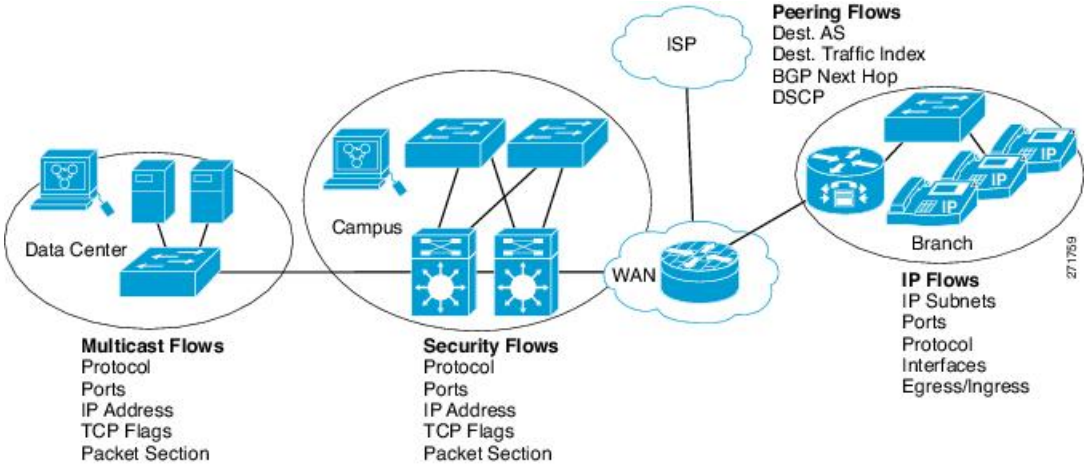
Original NetFlow allows you to understand the activities in the network and thus to optimize network design and reduce operational costs.

Flexible NetFlow allows you to understand network behavior with more efficiency, with specific flow information tailored for various services used in the network. The following are some example applications for a Flexible NetFlow feature:

- Flexible NetFlow enhances Cisco NetFlow as a security monitoring tool. For instance, new flow keys can be defined for packet length or MAC address, allowing users to search for a specific type of attack in the network.
- Flexible NetFlow allows you to quickly identify how much application traffic is being sent between hosts by specifically tracking TCP or UDP applications by the class of service (CoS) in the packets.
- The accounting of traffic entering a Multiprotocol Label Switching (MPLS) or IP core network and its destination for each next hop per class of service. This capability allows the building of an edge-to-edge traffic matrix.

The figure below is an example of how Flexible NetFlow might be deployed in a network.

Figure 2: Typical Deployment for Flexible NetFlow



Flexible NetFlow Components

Flexible NetFlow consists of components that can be used together in several variations to perform traffic analysis and data export. The user-defined flow records and the component structure of Flexible NetFlow facilitates the creation of various configurations for traffic analysis and data export on a networking device with a minimum number of configuration commands. Each flow monitor can have a unique combination of flow record, flow exporter, and cache type. If you change a parameter such as the destination IP address for a flow exporter, it is automatically changed for all the flow monitors that use the flow exporter. The same flow monitor can be used in conjunction with different flow samplers to sample the same type of network traffic at different rates on different interfaces. The following sections provide more information on Flexible NetFlow components:

Note Starting from Cisco IOS XE Release 3.10S, the number of configurable flow record fields have been increased from 32 to 40.

Flow Records

In Flexible NetFlow a combination of key and non-key fields is called a *flow record*. Flexible NetFlow records are assigned to Flexible NetFlow flow monitors to define the cache that is used for storing flow data. Flexible NetFlow includes several predefined records that can help you get started using Flexible NetFlow.

To use Flexible NetFlow to its fullest potential, you need to create your own customized records, as described in the following section(s):

NetFlow Predefined Records

Flexible NetFlow includes several predefined records that you can use to start monitoring traffic in your network. The predefined records are available to help you quickly deploy Flexible NetFlow and are easier to use than user-defined flow records. You can choose from a list of already defined records that may meet the needs for network monitoring. As Flexible NetFlow evolves, popular user-defined flow records will be made available as predefined records to make them easier to implement.

The predefined records ensure backward compatibility with your existing NetFlow collector configurations for the data that is exported. Each of the predefined records has a unique combination of key and nonkey fields that offer you the built-in ability to monitor various types of traffic in your network without customizing Flexible NetFlow on your router.

Two of the predefined records (NetFlow original and NetFlow IPv4/IPv6 original output), which are functionally equivalent, emulate original (ingress) NetFlow and the Egress NetFlow Accounting feature in original NetFlow, respectively. Some of the other Flexible NetFlow predefined records are based on the aggregation cache schemes available in original NetFlow. The Flexible NetFlow predefined records that are based on the aggregation cache schemes available in original NetFlow do not perform aggregation. Instead each flow is tracked separately by the predefined records.

User-Defined Records

Flexible NetFlow enables you to define your own records for a Flexible NetFlow flow monitor cache by specifying the key and nonkey fields to customize the data collection to your specific requirements. When you define your own records for a Flexible NetFlow flow monitor cache, they are referred to as *user-defined records*. The values in nonkey fields are added to flows to provide additional information about the traffic in the flows. A change in the value of a nonkey field does not create a new flow. In most cases the values for nonkey fields are taken from only the first packet in the flow. Flexible NetFlow enables you to capture counter values such as the number of bytes and packets in a flow as nonkey fields.

You can create user-defined records for applications such as QoS and bandwidth monitoring, application and end user traffic profiling, and security monitoring for dDoS attacks. Flexible NetFlow also includes several predefined records that emulate original NetFlow. Flexible NetFlow user-defined records provide the capability to monitor a contiguous section of a packet of a user-configurable size, and use it in a flow record as a key or a nonkey field along with other fields and attributes of the packet. The section may include any Layer 3 data from the packet. The packet section fields allow the user to monitor any packet fields that are not covered by the Flexible NetFlow predefined keys. The ability to analyze packet fields that are not collected with the predefined keys enables more detailed traffic monitoring, facilitates the investigation of dDoS attacks, and enables implementation of other security applications such as URL monitoring.

Flexible NetFlow provides predefined types of packet sections of a user-configurable size. The following Flexible NetFlow commands (used in Flexible NetFlow flow record configuration mode) can be used to configure the predefined types of packet sections:

- **collect ipv4 section header size** *bytes* --Starts capturing the number of bytes specified by the *bytes*argument from the beginning of the IPv4 header of each packet.
- **collect ipv4 section payload size** *bytes* --Starts capturing bytes immediately after the IPv4 header from each packet. The number of bytes captured is specified by the *bytes* argument.
- **collect ipv6 section header size** *bytes* --Starts capturing the number of bytes specified by the *bytes*argument from the beginning of the IPv6 header of each packet.
- **collect ipv6 section payload size** *bytes* --Starts capturing bytes immediately after the IPv6 header from each packet. The number of bytes captured is specified by the *bytes* argument.

The *bytes* values are the sizes in bytes of these fields in the flow record. If the corresponding fragment of the packet is smaller than the requested section size, Flexible NetFlow will fill the rest of the section field in the flow record with zeros. If the packet type does not match the requested section type, Flexible NetFlow will fill the entire section field in the flow record with zeros.

Flexible NetFlow adds a new Version 9 export format field type for the header and packet section types. Flexible NetFlow will communicate to the NetFlow collector the configured section sizes in the corresponding Version 9 export template fields. The payload sections will have a corresponding length field that can be used to collect the actual size of the collected section.

Note In Cisco IOS Release 12.2(50)SY, packet sections and payloads are not supported.

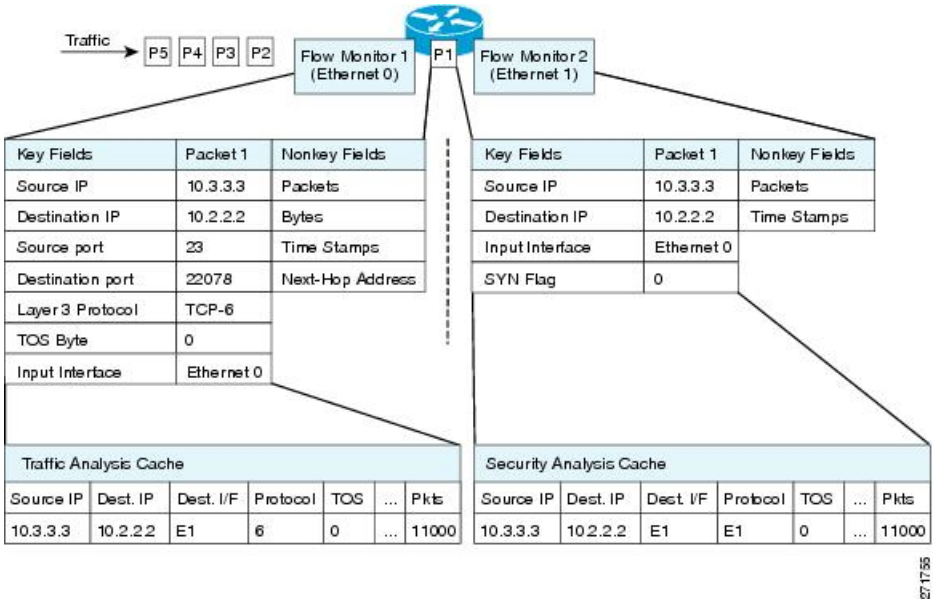
Flow Monitors

Flow monitors are the Flexible NetFlow component that is applied to interfaces to perform network traffic monitoring.

Flow data is collected from the network traffic and added to the flow monitor cache during the monitoring process based on the key and nonkey fields in the flow record.

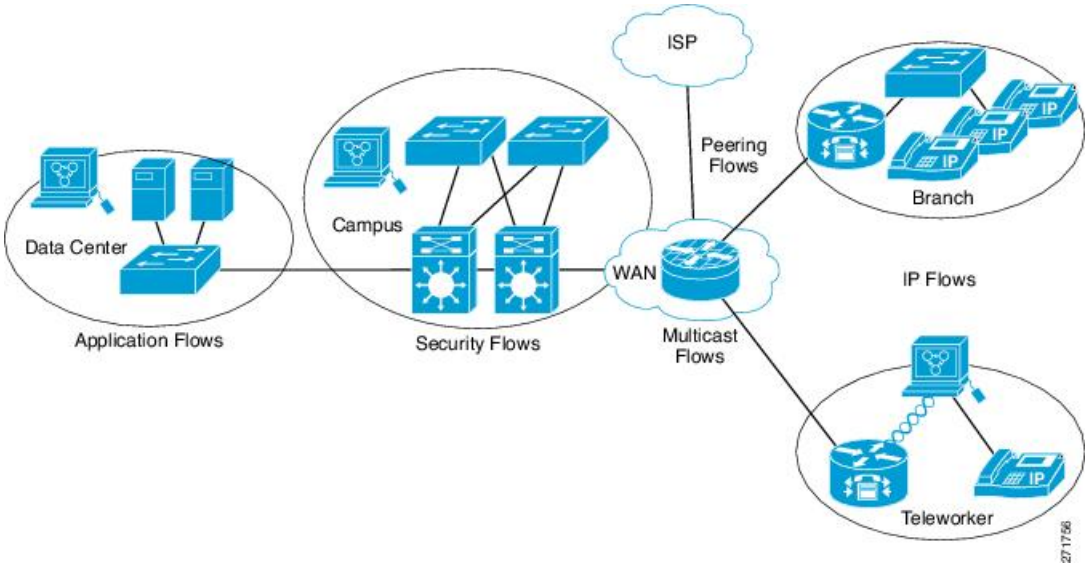
Flexible NetFlow can be used to perform different types of analysis on the same traffic. In the figure below, packet 1 is analyzed using a record designed for standard traffic analysis on the input interface and a record designed for security analysis on the output interface.

Figure 3: Example of Using Two Flow Monitors to Analyze the Same Traffic



The figure below shows a more complex example of how you can apply different types of flow monitors with custom records.

Figure 4: Complex Example of Using Multiple Types of Flow Monitors with Custom Records



There are three types of flow monitor caches. You change the type of cache used by the flow monitor after you create the flow monitor. The three types of flow monitor caches are described in the following sections:

Normal

The default cache type is “normal”. In this mode, the entries in the cache are aged out according to the timeout active and timeout inactive settings. When a cache entry is aged out, it is removed from the cache and exported via any exporters configured.

Immediate

A cache of type "immediate" ages out every record as soon as it is created. As a result, every flow contains just one packet. The commands that display the cache contents will provide a history of the packets seen.

This mode is desirable when you expect only very small flows and you want a minimum amount of latency between seeing a packet and exporting a report.

Caution This mode may result in a large amount of export data that can overload low-speed links and overwhelm any systems that you are exporting to. We recommended that you configure sampling to reduce the number of packets that are processed.

Note The cache timeout settings have no effect in this mode.

Permanent

A cache of type "permanent" never ages out any flows. A permanent cache is useful when the number of flows you expect to see is low and there is a need to keep long-term statistics on the router. For example, if the only key field in the flow record is the 8-bit IP ToS field, only 256 flows can be monitored. To monitor the long-term usage of the IP ToS field in the network traffic, you can use a permanent cache. Permanent caches are useful for billing applications and for an edge-to-edge traffic matrix for a fixed set of flows that are being tracked. Update messages will be sent periodically to any flow exporters configured according to the "timeout update" setting.

Note When a cache becomes full in permanent mode, new flows will not be monitored. If this occurs, a "Flows not added" message will appear in the cache statistics.

Note A permanent cache uses update counters rather than delta counters. This means that when a flow is exported, the counters represent the totals seen for the full lifetime of the flow and not the additional packets and bytes seen since the last export was sent.

Flow Exporters

Flow exporters export the data in the flow monitor cache to a remote system, such as a server running NetFlow collector, for analysis and storage. Flow exporters are created as separate entities in the configuration. Flow exporters are assigned to flow monitors to provide data export capability for the flow monitors. You can create several flow exporters and assign them to one or more flow monitors to provide several export destinations. You can create one flow exporter and apply it to several flow monitors.

NetFlow Data Export Format Version 9

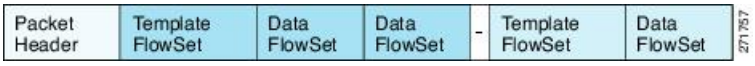
The basic output of NetFlow is a flow record. Several different formats for flow records have evolved as NetFlow has matured. The most recent evolution of the NetFlow export format is known as Version 9. The distinguishing feature of the NetFlow Version 9 export format is that it is template-based. Templates provide

an extensible design to the record format, a feature that should allow future enhancements to NetFlow services without requiring concurrent changes to the basic flow-record format. Using templates provides several key benefits:

- Third-party business partners who produce applications that provide collector or display services for NetFlow do not have to recompile their applications each time a new NetFlow feature is added. Instead, they should be able to use an external data file that documents the known template formats.
- New features can be added to NetFlow quickly without breaking current implementations.
- NetFlow is "future-proofed" against new or developing protocols because the Version 9 format can be adapted to provide support for them.

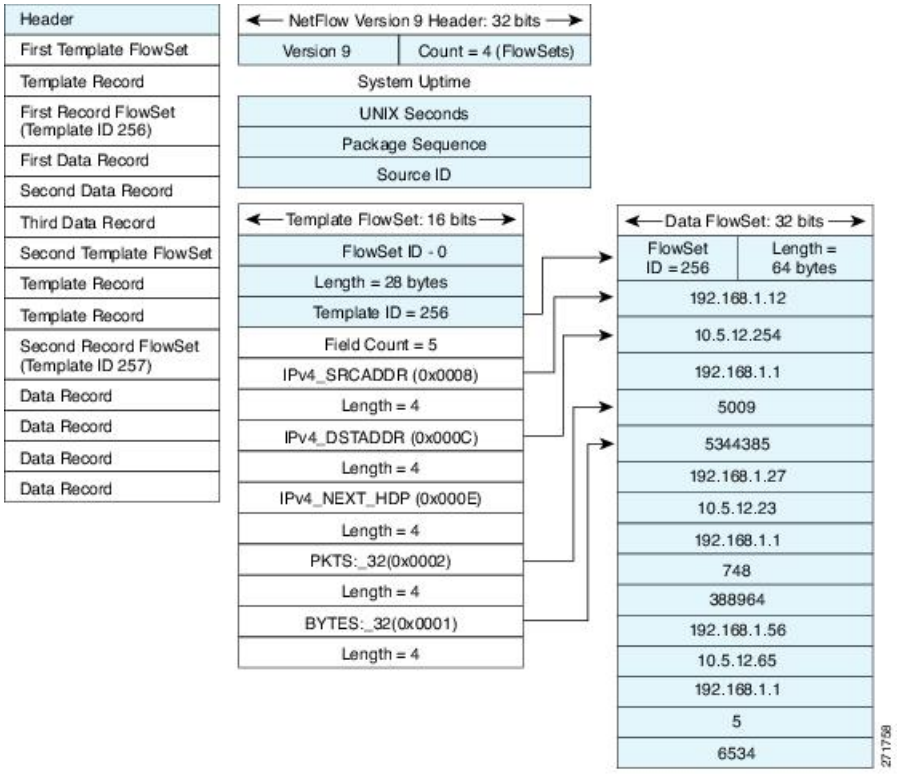
The Version 9 export format consists of a packet header followed by one or more template flow or data flow sets. A template flow set provides a description of the fields that will be present in future data flow sets. These data flow sets may occur later within the same export packet or in subsequent export packets. Template flow and data flow sets can be intermingled within a single export packet, as illustrated in the figure below.

Figure 5: Version 9 Export Packet



NetFlow Version 9 will periodically export the template data so the NetFlow collector will understand what data is to be sent and also export the data flow set for the template. The key advantage to Flexible NetFlow is that the user configures a flow record, which is effectively converted to a Version 9 template and then forwarded to the collector. The figure below is a detailed example of the NetFlow Version 9 export format, including the header, template flow, and data flow sets.

Figure 6: Detailed Example of the NetFlow Version 9 Export Format



For more information on the Version 9 export format, refer to the white paper titled Cisco IOS NetFlow Version 9 Flow-Record Format, available at this URL: http://www.cisco.com/en/US/tech/tk648/tk362/technologies_white_paper09186a00800a3db9.shtml.

Flow Samplers

Flow samplers are created as separate components in a router's configuration. Flow samplers are used to reduce the load on the device that is running by limiting the number of packets that are selected for analysis.

Flow sampling exchanges monitoring accuracy for router performance. When you apply a sampler to a flow monitor, the overhead load on the router of running the flow monitor is reduced because the number of packets that the flow monitor must analyze is reduced. The reduction in the number of packets that are analyzed by the flow monitor causes a corresponding reduction in the accuracy of the information stored in the flow monitor's cache.

Samplers are combined with flow monitors when they are applied to an interface with the **ip flow monitor** command.

Security Monitoring with Flexible NetFlow

Flexible NetFlow can be used as a network attack detection tool with capabilities to track all parts of the IP header and even packet sections and characterize this information into flows. Security monitoring systems can analyze Flexible NetFlow data, and upon finding an issue in the network, create a virtual bucket or virtual cache that will be configured to track specific information and identify details about the attack pattern or worm

propagation. The capability to create caches dynamically with specific information combined with input filtering (for example, filtering all flows to a specific destination) makes Flexible NetFlow a powerful security monitoring tool.

One common type of attack occurs when TCP flags are used to flood open TCP requests to a destination server (for example, a SYN flood attack). The attacking device sends a stream of TCP SYNs to a given destination address but never sends the ACK in response to the servers SYN-ACK as part of the TCP three-way handshake. The flow information needed for a security detection server requires the tracking of three key fields: destination address or subnet, TCP flags, and packet count. The security detection server may be monitoring general Flexible NetFlow information, and this data may trigger a detailed view of this particular attack by the Flexible NetFlow dynamically creating a new flow monitor in the router's configuration. The new flow monitor might include input filtering to limit what traffic is visible in the Flexible NetFlow cache along with the tracking of the specific information to diagnose the TCP-based attack. In this case the user may want to filter all flow information to the server destination address or subnet to limit the amount of information the security detection server needs to evaluate. If the security detection server decided it understood this attack, it might then program another flow monitor to collect and export payload information or sections of packets to take a deeper look at a signature within the packet. This example is just one of many possible ways that Flexible NetFlow can be used to detect security incidents.

Feature Comparison of Original NetFlow and Flexible NetFlow

The table below provides a feature-by-feature comparison of original NetFlow and Flexible NetFlow.

Table 1: Feature-by-Feature Comparison of Original NetFlow and Flexible NetFlow

Feature	Original NetFlow	Flexible NetFlow	Comments
NetFlow Data Capture	Supported	Supported	Data capture is available with the predefined and user-defined records in Flexible NetFlow. Flexible NetFlow has several predefined keys that emulate the traffic analysis capabilities of original NetFlow.
NetFlow Data Export	Supported	Supported	Flow exporters export data from the Flexible NetFlow flow monitor caches to remote systems.
NetFlow for IPv6	Supported	Supported	IPv6 support was removed from original NetFlow in Cisco IOS Release 12.4(20)T. The Flexible NetFlow--IPv6 Unicast Flows feature implemented IPv6 support for Flexible NetFlow in Cisco IOS Release 12.4(20)T.
NetFlow BGP Next Hop Support	Supported	Supported	Available in the predefined and user-defined keys in Flexible NetFlow records.
Random Packet Sampled NetFlow	Supported	Supported	Available with Flexible NetFlow sampling.

Feature	Original NetFlow	Flexible NetFlow	Comments
NetFlow v9 Export Format	Supported	Supported	Available with Flexible NetFlow exporters.
NetFlow Subinterface Support	Supported	Supported	Flexible NetFlow monitors can be assigned to subinterfaces.
NetFlow Multiple Export Destinations	Supported	Supported	Available with Flexible NetFlow exporters.
NetFlow ToS-Based Router Aggregation	Supported	Supported	Available in the predefined and user-defined records in Flexible NetFlow records.
NetFlow Minimum Prefix Mask for Router-Based Aggregation	Supported	Supported	Available in the predefined and user-defined records.
NetFlow Input Filters	Supported	Not supported	--
NetFlow MIB	Supported	Not supported	--
Egress NetFlow Accounting	Supported	Supported	Flexible NetFlow monitors can be used to monitor egress traffic on interfaces and subinterfaces.

Criteria for Identifying Traffic to Be Used in Analysis in Flexible NetFlow

If the predefined Flexible NetFlow records are not suitable for your traffic requirements, you can create a user-defined (custom) record using the Flexible NetFlow **collect** and **match** commands. Before you can create a customized record, you must decide the criteria that you are going to use for the key and nonkey fields.

If you want to create a customized record for detecting network attacks, you must include the appropriate key and nonkey fields in the record to ensure that the router creates the flows and captures the data that you need to analyze the attack and respond to it. For example, SYN flood attacks are a common denial of service (DoS) attack in which TCP flags are used to flood open TCP requests to a destination host. When a normal TCP connection starts, a destination host receives a SYN (synchronize/start) packet from a source host and sends back a SYN ACK (synchronize acknowledge). The destination host must then hear an ACK (acknowledge) of the SYN ACK before the connection is established. This is referred to as the "TCP three-way handshake While the destination host waits for the ACK to the SYN ACK, a connection queue of finite size on the destination host keeps track of connections waiting to be completed. This queue typically empties quickly because the ACK is expected to arrive a few milliseconds after the SYN ACK. The TCP SYN attack exploits this design by having an attacking source host generate TCP SYN packets with random source addresses toward a victim host. The victim destination host sends a SYN ACK back to the random source address and adds an entry to the connection queue. Because the SYN ACK is destined for an incorrect or nonexistent host, the last part of the TCP three-way handshake is never completed and the entry remains in the connection queue until a timer expires, typically for about one minute. When the source host rapidly generates TCP SYN packets from random IP addresses, the connection queue can be filled and TCP services (such as e-mail, file transfer, or WWW) can be denied to legitimate users.

The information needed for a security monitoring record for this type of DoS attack might include the following key and nonkey fields:

- Key fields:
  - Destination IP address or destination IP subnet
  - TCP flags
  - Packet count
- Nonkey fields
  - Destination IP address
  - Source IP address
  - Interface input and output

Tip Many users configure a general Flexible NetFlow monitor that triggers a more detailed Flexible NetFlow view of a DoS attack using these key and nonkey fields.

Benefit of Emulating Original NetFlow with Flexible NetFlow

Emulating original NetFlow with Flexible NetFlow enables to you to deploy Flexible NetFlow quickly because you can use a predefined record instead of designing and configuring a custom user-defined record. You need only configure a flow monitor and apply it to an interface for Flexible NetFlow to start working like original NetFlow. You can add an optional exporter if you want to analyze the data that you collect with an application such as NetFlow collector.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. The record format can be one of the predefined record formats, or an advanced user may create his or her own record format using the **collect** and **match**commands in Flexible NetFlow flow record configuration mode.

Flow exporters are used to send the data that you collect with Flexible NetFlow to a remote system such as a NetFlow Collection Engine. Exporters use UDP as the transport protocol and use the Version 9 export format.

If you are familiar with original NetFlow, you already understand the format and content of the data that you collect and export with Flexible NetFlow when you emulate original NetFlow. You will be able to use the same techniques for analyzing the data.

Flexible NetFlow Predefined Records

Flexible NetFlow predefined records are based on the original NetFlow ingress and egress caches and the aggregation caches. The difference between the original NetFlow aggregation caches and the corresponding predefined Flexible NetFlow records is that the predefined records do not perform aggregation. Flexible NetFlow predefined records are associated with a Flexible NetFlow flow monitor the same way that you associate a user-defined (custom) record.

Benefits of Flexible NetFlow Predefined Records

If you have been using original NetFlow or original NetFlow with aggregation caches you can continue to capture the same traffic data for analysis when you migrate to Flexible NetFlow by using the predefined records available with Flexible NetFlow. Many users will find that the preexisting Flexible NetFlow records are suitable for the majority of their traffic analysis requirements.

NetFlow Original and NetFlow IPv4 Original Input Predefined Records

The Flexible NetFlow "NetFlow original" and "NetFlow IPv4 original input" predefined records can be used interchangeably because they have the same key and nonkey fields. The key and nonkey fields and the counters for the Flexible NetFlow "NetFlow original" and "NetFlow IPv4 original input" predefined records are shown in the table below.

Table 2: Key and Nonkey Fields Used by the Flexible NetFlow NetFlow Original and NetFlow IPv4 Original Input Predefined Records

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the type of service (ToS) field.
IP Protocol	Key	Value in the IP protocol field.
IP Source Address	Key	IP source address.
IP Destination Address	Key	IP destination address.
Transport Source Port	Key	Value of the transport layer source port field.
Transport Destination Port	Key	Value of the transport layer destination port field.
Interface Input	Key	Interface on which the traffic is received.
Flow Sampler ID	Key	ID number of the flow sampler (if flow sampling is enabled).
IP Source AS	Nonkey	Source autonomous system number.
IP Destination AS	Nonkey	Destination autonomous system number.
IP Next Hop Address	Nonkey	IP address of the next hop.
IP Source Mask	Nonkey	Mask for the IP source address.
IP Destination Mask	Nonkey	Mask for the IP destination address.
TCP Flags	Nonkey	Value in the TCP flag field.
Interface Output	Nonkey	Interface on which the traffic is transmitted.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

NetFlow IPv4 Original Output Predefined Record

The Flexible NetFlow "NetFlow IPv4 original output" predefined record is used to emulate the original NetFlow Egress NetFlow Accounting feature that was released in Cisco IOS Release 12.3(11)T. The key and nonkey fields and the counters for the Flexible NetFlow "NetFlow IPv4 original output" predefined record are shown in the table below.

Table 3: Key and Nonkey Fields Used by the Flexible NetFlow NetFlow IPv4 Original Output Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Protocol	Key	Value in the IP protocol field.
IP Source Address	Key	IP source address.
IP Destination Address	Key	IP destination address.
Transport Source Port	Key	Value of the transport layer source port field.
Transport Destination Port	Key	Value of the transport layer destination port field.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Sampler ID	Key	ID number of the flow sampler (if flow sampling is enabled).
IP Source AS	Nonkey	Source autonomous system number.
IP Destination AS	Nonkey	Destination autonomous system number.
IP Next Hop Address	Nonkey	IP address of the next hop.
IP Source Mask	Nonkey	Mask for the IP source address.
IP Destination Mask	Nonkey	Mask for the IP destination address.
TCP Flags	Nonkey	Value in the TCP flag field.
Interface Input	Nonkey	Interface on which the traffic is received.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

The configuration in the Example: Configuring Flexible NetFlow Egress Accounting for IPv4 and IPv6 Traffic, on page 48 uses the predefined Flexible NetFlow "NetFlow original output" record.

NetFlow IPv6 Original Input Predefined Record

The key and nonkey fields and the counters for the Flexible NetFlow "NetFlow IPv6 original input" predefined record are shown in the table below.

Table 4: Key and Nonkey Fields Used by the Flexible NetFlow NetFlow IPv6 Original Input Predefined Record

Field	Key or NonKey Field	Definition
Traffic Class	Key	Value in the traffic class field.
Flow Label	Key	Flow label.
Protocol	Key	Value in the protocol field.
Extension Map	Key	Value in the extension map bitmap.
IP Source Address	Key	IP source address.
IP Destination Address	Key	IP destination address.
Transport Source Port	Key	Value of the transport layer source port field.
Transport Destination Port	Key	Value of the transport layer destination port field.
Interface Input	Key	Interface on which the traffic is received.
Flow Direction	Key	The direction of the flow.
Flow Sampler	Key	ID number of the flow sampler (if flow sampling is enabled).
Routing Source AS	Nonkey	Source autonomous system number.
Routing Destination AS	Nonkey	Destination autonomous system number.
Routing Next-hop Address	Nonkey	IP address of the next hop.
IP Source Mask	Nonkey	Mask for the IP source address.
IP Destination Mask	Nonkey	Mask for the IP destination address.
Transport TCP Flags	Nonkey	Value in the TCP flag field.
Interface Output	Nonkey	Interface over which the traffic is transmitted.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.

Field	Key or NonKey Field	Definition
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

NetFlow IPv6 Original Output Predefined Record

The key and nonkey fields and the counters for the Flexible NetFlow "NetFlow IPv6 original output" predefined record are shown in the table below.

Table 5: Key and Nonkey Fields Used by the Flexible NetFlow NetFlow IPv6 Original Output Predefined Record

Field	Key or Nonkey Field	Definition
Traffic Class	Key	Value in the traffic class field.
Flow Label	Key	The flow label.
Protocol	Key	Value in the protocol field.
Extension Map	Key	Value in the extension map bitmap.
IP Source Address	Key	IP source address.
IP Destination Address	Key	IP destination address.
Transport Source Port	Key	Value of the transport layer source port field.
Transport Destination Port	Key	Value of the transport layer destination port field.
Interface Output	Key	Interface over which the traffic is transmitted.
Flow Direction	Key	The direction of the flow.
Flow Sampler	Key	ID number of the flow sampler (if flow sampling is enabled).
Routing Source AS	Nonkey	Source autonomous system number.
Routing Destination AS	Nonkey	Destination autonomous system number.
Routing Next-hop Address	Nonkey	IP address of the next hop.
IP Source Mask	Nonkey	Mask for the IP source address.
IP Destination Mask	Nonkey	Mask for the IP destination address.
Transport TCP Flags	Nonkey	Value in the TCP flag field.
Interface Input	Nonkey	Interface on which the traffic is received.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.

Field	Key or Nonkey Field	Definition
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Autonomous System Predefined Record

The Flexible NetFlow "autonomous system" predefined record creates flows based on autonomous system-to-autonomous system traffic flow data. The Flexible NetFlow "autonomous system" predefined record uses the same key and nonkey fields as the original NetFlow "autonomous system" aggregation cache.

Note This predefined record can be used to analyze IPv4 and IPv6 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "autonomous system" predefined record.

Table 6: Key and Nonkey Fields Used by the Flexible NetFlow Autonomous System Predefined Record

Field	Key or Nonkey Field	Definition
IP Source AS	Key	Autonomous system of the source IP address (peer or origin).
IP Destination AS	Key	Autonomous system of the destination IP address (peer or origin).
Interface Input	Key	Interface on which the traffic is received.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds since this device was first booted) when the last packet was switched.

Autonomous System ToS Predefined Record

The Flexible NetFlow "autonomous system ToS" predefined record creates flows based on autonomous system-to-autonomous system and type of service (ToS) traffic flow data. The Flexible NetFlow "autonomous

system ToS" predefined record uses the same key and nonkey fields as the original NetFlow "autonomous system ToS" aggregation cache.

Note This predefined record can be used to analyze only IPv4 traffic.

Tip This predefined record is particularly useful for generating autonomous system-to-autonomous system traffic flow data.

The table below lists the key and nonkey fields used in the Flexible NetFlow "autonomous system ToS" predefined record.

Table 7: Key and Nonkey Fields Used by the Flexible NetFlow Autonomous System ToS Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Source autonomous system	Key	Autonomous system of the source IP address (peer or origin).
IP Destination autonomous system	Key	Autonomous system of the destination IP address (peer or origin).
Interface Input	Key	Interface on which the traffic is received.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

BGP Next-Hop Predefined Record

The Flexible NetFlow "BGP next-hop" predefined record creates flows based on Border Gateway Protocol (BGP) traffic flow data.

Note This predefined record can be used to analyze only IPv6 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "BGP next-hop" predefined record.

Table 8: Key and Nonkey Fields Used by the Flexible NetFlow BGP Next-Hop Predefined Record

Field	Key or Nonkey Field	Definition
Routing Source AS	Key	Autonomous system of the source IP address.
Routing Destination AS	Key	Autonomous system of the destination IP address.
Routing Next-hop Address IPv6 BGP	Key	IPv6 address of the BGP next hop.
Interface Input	Key	Interface on which the traffic is received.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Timestamp Sys-uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Timestamp Sys-uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

BGP Next-Hop ToS Predefined Record

The Flexible NetFlow "BGP next-hop ToS" predefined record creates flows based on BGP and ToS traffic flow data. The Flexible NetFlow "BGP next-hop ToS" predefined record uses the same key and nonkey fields as the original NetFlow "BGP next-hop ToS" aggregation cache.

Note This predefined record can be used to analyze only IPv4 traffic.

The table below lists the key and nonkey fields used in the "BGP next-hop ToS" predefined record.

Table 9: Key and Nonkey Fields Used by the Flexible NetFlow BGP Next-Hop ToS Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Source autonomous system	Key	Autonomous system of the source IP address (peer or origin).

Field	Key or Nonkey Field	Definition
IP Destination autonomous system	Key	Autonomous system of the destination IP address (peer or origin).
IPv4 Next Hop Address BGP	Key	IPv4 address of the BGP next hop.
Interface Input	Key	Interface on which the traffic is received.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Destination Prefix Predefined Record

The Flexible NetFlow "destination prefix" predefined record creates flows based on destination prefix traffic flow data. The Flexible NetFlow "destination prefix" predefined record uses the same key and nonkey fields as the original NetFlow "destination prefix" aggregation cache.

Note This predefined record can be used to analyze IPv4 and IPv6 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "destination prefix" predefined record.

Table 10: Key and Nonkey Fields Used by the Flexible NetFlow Destination Prefix Predefined Record

Field	Key or Nonkey Field	Definition
IP Destination autonomous system	Key	Autonomous system of the destination IP address (peer or origin).
IPv4 or IPv6 Destination Prefix	Key	Destination IP address ANDed with the destination prefix mask.
IPv4 or IPv6 Destination Mask	Key	Number of bits in the destination prefix.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Direction	Key	Direction in which the flow is being monitored.

Field	Key or Nonkey Field	Definition
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Destination Prefix ToS Predefined Record

The Flexible NetFlow "destination prefix ToS" predefined record creates flows based on destination prefix and ToS traffic flow data. The Flexible NetFlow "destination prefix ToS" predefined record uses the same key and nonkey fields as the original NetFlow "destination prefix ToS" aggregation cache.

This predefined record is particularly useful for capturing data with which you can examine the destinations of network traffic passing through a NetFlow-enabled device.

Note This predefined record can be used to analyze only IPv4 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "destination prefix ToS" predefined record.

Table 11: Key and Nonkey Fields Used by the Flexible NetFlow Destination Prefix ToS Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Destination autonomous system	Key	Autonomous system of the destination IP address (peer or origin).
IPv4 Destination Prefix	Key	Destination IP address ANDed with the destination prefix mask.
IPv4 Destination Mask	Key	Number of bits in the destination prefix.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.

Field	Key or Nonkey Field	Definition
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Prefix Predefined Record

The Flexible NetFlow "prefix" predefined record creates flows based on the source and destination prefixes in the traffic flow data. The Flexible NetFlow "prefix" predefined record uses the same key and nonkey fields as the original NetFlow "prefix" aggregation cache.

Note This predefined record can be used to analyze IPv4 and IPv6 traffic. For IPv6 traffic, a minimum prefix mask length of 0 bits is assumed.

The table below lists the key and nonkey fields used in the Flexible NetFlow "prefix" predefined record.

Table 12: Key and Nonkey Fields Used by the Flexible NetFlow Prefix Predefined Record

Field	Key or Nonkey Field	Definition
IP Source autonomous system	Key	Autonomous system of the source IP address (peer or origin).
IP Destination autonomous system	Key	Autonomous system of the destination IP address (peer or origin).
IPv4 or IPv6 Source Prefix	Key	Source IP address ANDed with the source prefix mask, or the prefix to which the source IP address of the aggregated flows belongs.
IPv4 or IPv6 Source Mask	Key	Number of bits in the source prefix.
IPv4 or IPv6 Destination Prefix	Key	Destination IP address ANDed with the destination prefix mask.
IPv4 or IPv6 Destination Mask	Key	Number of bits in the destination prefix.
Interface Input	Key	Interface on which the traffic is received.
Interface Output	Key	Interface on which the traffic is transmitted.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.

Field	Key or Nonkey Field	Definition
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Prefix Port Predefined Record

The Flexible NetFlow "prefix port" predefined record creates flows based on source and destination prefixes and ports in the traffic flow data. The Flexible NetFlow "prefix port" predefined record uses the same key and nonkey fields as the original NetFlow "prefix port" aggregation cache.

This predefined record is particularly useful for capturing data with which you can examine the sources and destinations of network traffic passing through a NetFlow-enabled device.

Note This predefined record can be used to analyze only IPv4 traffic.

The table below lists the key and nonkey fields used in the destination Flexible NetFlow "prefix port" predefined record.

Table 13: Key and Nonkey Fields Used by the Flexible NetFlow Prefix Port Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Protocol	Key	Value in the IP protocol field.
IPv4 Source Prefix	Key	Source IP address ANDed with the source prefix mask, or the prefix to which the source IP address of the aggregated flows belongs.
IPv4 Source Mask	Key	Number of bits in the source prefix.
IPv4 Destination Prefix	Key	Destination IP address ANDed with the destination prefix mask.
IPv4 Destination Mask	Key	Number of bits in the destination prefix.
Transport Source Port	Key	Value in the transport layer source port field.
Transport Destination Port	Key	Value in the transport layer destination port field.
Interface Input	Key	Interface on which the traffic is received.
Interface Output	Key	Interface on which the traffic is transmitted.

Field	Key or Nonkey Field	Definition
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Prefix ToS Predefined Record

The Flexible NetFlow "prefix ToS" predefined record creates flows based on source and destination prefixes and ToS traffic flow data. The Flexible NetFlow "prefix ToS" predefined record uses the same key and nonkey fields as the original NetFlow "destination prefix ToS" aggregation cache.

This predefined record is particularly useful for capturing data with which you can examine the sources and destinations of network traffic passing through a NetFlow-enabled device.

Note This predefined record can be used to analyze only IPv4 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "prefix ToS" predefined record.

Table 14: Key and Nonkey Fields Used by the Flexible NetFlow Prefix ToS Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Source autonomous system	Key	Autonomous system of the source IP address (peer or origin).
IP Destination autonomous system	Key	Autonomous system of the destination IP address (peer or origin).
IPv4 Source Prefix	Key	Source IP address ANDed with the source prefix mask, or the prefix to which the source IP address of the aggregated flows belongs.
IPv4 Source Mask	Key	Number of bits in the source prefix.
IPv4 Destination Prefix	Key	Destination IP address ANDed with the destination prefix mask.
IPv4 Destination Mask	Key	Number of bits in the destination prefix.

Field	Key or Nonkey Field	Definition
Interface Input	Key	Interface on which the traffic is received.
Interface Output	Key	Interface on which the traffic is transmitted.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Protocol Port Predefined Record

The Flexible NetFlow "protocol port" predefined record creates flows based on protocols and ports in the traffic flow data. The Flexible NetFlow "protocol port" predefined record uses the same key and nonkey fields as the original NetFlow "protocol port" aggregation cache.

Note This predefined record can be used to analyze IPv4 and IPv6 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "protocol port" predefined record.

Table 15: Key and Nonkey Fields Used by the Flexible NetFlow Protocol Port Predefined Record

Field	Key or Nonkey Field	Definition
IP Protocol	Key	Value in the IP protocol field.
Transport Source Port	Key	Value in the transport layer source port field.
Transport Destination Port	Key	Value in the transport layer destination port field.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.

Field	Key or Nonkey Field	Definition
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Protocol Port ToS Predefined Record

The Flexible NetFlow "protocol port ToS" predefined record creates flows based on the protocol, port, and ToS value in the traffic data. The Flexible NetFlow "protocol port ToS" predefined record uses the same key and nonkey fields as the original NetFlow "protocol port ToS" aggregation cache.

This predefined record is particularly useful for capturing data with which you can examine network usage by type of traffic.

Note This predefined record can be used to analyze only IPv4 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "protocol port ToS" predefined record.

Table 16: Key and Nonkey Fields Used by the Flexible NetFlow Protocol Port ToS Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Protocol	Key	Value in the IP protocol field.
Transport Source Port	Key	Value in the transport layer source port field.
Transport Destination Port	Key	Value in the transport layer destination port field.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Source Prefix Predefined Record

The Flexible NetFlow "source prefix" predefined record creates flows based on source prefixes in the network traffic. The Flexible NetFlow "source prefix" predefined record uses the same key and nonkey fields as the original NetFlow "source prefix" aggregation cache.

Note This predefined record can be used to analyze IPv4 and IPv6 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "source prefix" predefined record.

Table 17: Key and Nonkey Fields Used by the Flexible NetFlow Source Prefix Predefined Record

Field	Key or Nonkey Field	Definition
IP Source autonomous system	Key	Autonomous system of the source IP address (peer or origin).
IPv4 or IPv6 Source Prefix	Key	Source IP address ANDed with the source prefix mask, or the prefix to which the source IP address of the aggregated flows belongs.
IPv4 or IPv6 Source Mask	Key	Number of bits in the source prefix.
Interface Input	Key	Interface on which the traffic is received.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

Source Prefix ToS Predefined Record

The Flexible NetFlow "source prefix ToS" predefined record creates flows based on source prefixes and ToS values in the network traffic. The Flexible NetFlow "source prefix ToS" predefined record uses the same key and nonkey fields as the original NetFlow "source prefix ToS" aggregation cache.

This predefined record is particularly useful for capturing data with which you can examine the sources of network traffic passing through a NetFlow-enabled device.

Note This predefined record can be used to analyze only IPv4 traffic.

The table below lists the key and nonkey fields used in the Flexible NetFlow "source prefix ToS" predefined record.

Table 18: Key and Nonkey Fields Used by the Flexible NetFlow Source Prefix ToS Predefined Record

Field	Key or Nonkey Field	Definition
IP ToS	Key	Value in the ToS field.
IP Source autonomous system	Key	Autonomous system of the source IP address (peer or origin).
IPv4 Source Prefix	Key	Source IP address ANDed with the source prefix mask, or the prefix to which the source IP address of the aggregated flows belongs.
IPv4 Source Mask	Key	Number of bits in the source prefix.
Interface Input	Key	Interface on which the traffic is received.
Flow Direction	Key	Direction in which the flow is being monitored.
Counter Bytes	Nonkey	Number of bytes seen in the flow.
Counter Packets	Nonkey	Number of packets seen in the flow.
Time Stamp System Uptime First	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the first packet was switched.
Time Stamp System Uptime Last	Nonkey	System uptime (time, in milliseconds, since this device was first booted) when the last packet was switched.

How to Configure Flexible Netflow

Creating a Customized Flow Record

Perform this task to configure a customized flow record.

Customized flow records are used to analyze traffic data for a specific purpose. A customized flow record must have at least one **match** criterion for use as the key field and typically has at least one **collect** criterion for use as a nonkey field.

There are hundreds of possible permutations of customized flow records. This task shows the steps that are used to create one of the possible permutations. Modify the steps in this task as appropriate to create a customized flow record for your requirements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **description** *description*
5. **match** {**ip** | **ipv6**} {**destination** | **source**} **address**
6. Repeat Step 5 as required to configure additional key fields for the record.

7. **collect interface** {**input** | **output**}
8. Repeat the above step as required to configure additional nonkey fields for the record.
9. **end**
10. **show flow record** *record-name*
11. **show running-config flow record** *record-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-RECORD-1`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. • This command also allows you to modify an existing flow record.
Step 4	**description** *description* **Example:** `Device(config-flow-record)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow record.
Step 5	**match** {**ip** | **ipv6**} {**destination** | **source**} **address** **Example:** `Device(config-flow-record)# match ipv4 destination address`	Configures a key field for the flow record. **Note** This example configures the IPv4 destination address as a key field for the record. For information about the other key fields available for the **match ipv4** command, and the other **match** commands that are available to configure key fields.
Step 6	Repeat Step 5 as required to configure additional key fields for the record.	—
Step 7	**collect interface** {**input** | **output**} **Example:**	Configures the input interface as a nonkey field for the record. **Note** This example configures the input interface as a nonkey field for the record.
Step 8	Repeat the above step as required to configure additional nonkey fields for the record.	—

	Command or Action	Purpose
Step 9	**end** **Example:** `Device(config-flow-record)# end`	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 10	**show flow record** *record-name* **Example:** `Device# show flow record FLOW_RECORD-1`	(Optional) Displays the current status of the specified flow record.
Step 11	**show running-config flow record** *record-name* **Example:** `Device# show running-config flow record FLOW_RECORD-1`	(Optional) Displays the configuration of the specified flow record.

Displaying the Current Status of a Flow Record

Perform this optional task to display the current status of a flow record.

SUMMARY STEPS

1. **enable**
2. **show flow record**

DETAILED STEPS

Step 1 enable

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 show flow record

The **show flow record** command shows the current status of the flow monitor that you specify.

Example:

```
Device# show flow record

flow record FLOW-RECORD-2:
  Description:        Used for basic IPv6 traffic analysis
  No. of users:       1
  Total field space:  53 bytes
  Fields:
    match ipv6 destination address
    collect counter bytes
```

```
    collect counter packets
flow record FLOW-RECORD-1:
  Description:        Used for basic IPv4 traffic analysis
  No. of users:       1
  Total field space:  29 bytes
  Fields:
    match ipv4 destination address
    collect counter bytes
    collect counter packets
```

Verifying the Flow Record Configuration

Perform this optional task to verify the configuration commands that you entered.

SUMMARY STEPS

1. **enable**
2. **show running-config flow record**

DETAILED STEPS

Step 1 enable

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 show running-config flow record

The **show running-config flow record** command shows the configuration commands of the flow monitor that you specify.

Example:

```
Device# show running-config flow record

Current configuration:
!
flow record FLOW-RECORD-2
 description Used for basic IPv6 traffic analysis
 match ipv6 destination address
 collect counter bytes
 collect counter packets
!
flow record FLOW-RECORD-1
 description Used for basic IPv4 traffic analysis
 match ipv4 destination address
 collect counter bytes
 collect counter packets
```

!

Configuring a Flow Monitor for IPv4 or IPv6 Traffic Using the Predefined Record

To configure a flow monitor for IPv4/IPv6 traffic using the Flexible NetFlow "NetFlow IPv4/IPv6 original input" predefined record for the flow monitor, perform the following required task.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. The record format can be one of the predefined record formats, or an advanced user may create his or her own record format using the **collect** and **match**commands in Flexible NetFlow flow record configuration mode.

Note You must remove a flow monitor from all of the interfaces to which you have applied it before you can modify the **record** format of the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record netflow {ipv4 | ipv6} original-input**
6. **end**
7. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]]
8. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.

	Command or Action	Purpose
Step 4	**description** *description* **Example:** `Device(config-flow-monitor)# description Used for monitoring IPv4 traffic`	(Optional) Creates a description for the flow monitor.
Step 5	**record netflow {ipv4 | ipv6} original-input** **Example:** `Device(config-flow-monitor)# record netflow ipv4 original-input`	Specifies the record for the flow monitor.
Step 6	**end** **Example:** `Device(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 7	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]] **Example:** `Device# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.
Step 8	**show running-config flow monitor** *monitor-name* **Example:** `Device# show flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Configuring a Flow Exporter for the Flow Monitor

Perform this optional task to configure a flow exporter for the flow monitor in order to export the data that is collected by Flexible NetFlow to a remote system for further analysis and storage.

Flow exporters are used to send the data that you collect with Flexible NetFlow to a remote system such as a NetFlow Collection Engine. Exporters use UDP as the transport protocol and use the Version 9 export format.

Note Each flow exporter supports only one destination. If you want to export the data to multiple destinations, you must configure multiple flow exporters and assign them to the flow monitor.

You can export to a destination using either an IPv4 or IPv6 address.

Note When you configure an exporter, configure the exporter in such a way that the source interface is defined as a WAN interface. This configuration helps you prevent any unpredictable behavior because the NAT is not applied on the packets.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*hostname* | *ip-address*} [**vrf** *vrf-name*]
6. **export-protocol** {**netflow-v5** | **netflow-v9** | **ipfix**}
7. **transport udp** *udp-port*
8. **exit**
9. **flow monitor** *flow-monitor-name*
10. **exporter** *exporter-name*
11. **end**
12. **show flow exporter** *exporter-name*
13. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates a flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to datacenter`	(Optional) Creates a description for the flow exporter.
Step 5	**destination** {*hostname*	*ip-address*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the hostname or IP address of the system to which the exporter sends data. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**export-protocol** {**netflow-v5**	**netflow-v9**	**ipfix**} **Example:**	Specifies the version of the NetFlow export protocol used by the exporter.

	Command or Action	Purpose
	`Device(config-flow-exporter)# export-protocol netflow-v9`	• Default: **netflow-v9**.
Step 7	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 65`	Configures UDP as the transport protocol and specifies the UDP port on which the destination system is listening for exported Flexible NetFlow traffic.
Step 8	**exit** **Example:** `Device(config-flow-exporter)# exit`	Exits Flexible NetFlow flow exporter configuration mode and returns to global configuration mode.
Step 9	**flow monitor** *flow-monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Enters Flexible NetFlow flow monitor configuration mode for the flow monitor that you created previously.
Step 10	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter EXPORTER-1`	Specifies the name of an exporter that you created previously.
Step 11	**end** **Example:** `Device(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 13	**show running-config flow exporter** *exporter-name* **Example:** `Device<# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Creating a Customized Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be one of the predefined formats or a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record instead of using one of the Flexible NetFlow predefined records, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}}
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **statistics packet protocol**
9. **statistics packet size**
10. **exporter** *exporter-name*
11. **end**
12. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]]
13. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.

	Command or Action	Purpose
Step 4	**description** *description* **Example:** `(config-flow-monitor)# description Used for basic ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}} **Example:**	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. The values for the keywords associated with the **timeout** keyword have no effect when the cache type is set to **immediate**.
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	**statistics packet protocol** **Example:** `(config-flow-monitor)# statistics packet protocol`	(Optional) Enables the collection of protocol distribution statistics for Flexible NetFlow monitors.
Step 9	**statistics packet size** **Example:** `(config-flow-monitor)# statistics packet size`	(Optional) Enables the collection of size distribution statistics for Flexible NetFlow monitors.
Step 10	**exporter** *exporter-name* **Example:** `(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.
Step 11	**end** **Example:** `(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] **Example:** `# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.

	Command or Action	Purpose
Step 13	**show running-config flow monitor** *monitor-name* **Example:** `# show running-config flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Displaying the Current Status of a Flow Monitor

Perform this optional task to display the current status of a flow monitor.

SUMMARY STEPS

1. **enable**
2. **show flow monitor** *monitor-name*

DETAILED STEPS

Step 1 **enable**

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 **show flow monitor** *monitor-name*

The **show flow monitor** command shows the current status of the flow monitor that you specify.

Example:

```
Device# show flow monitor FLOW-MONITOR-1

Flow Monitor FLOW-MONITOR-1:
  Description:       Used for basic ipv4 traffic analysis
  Flow Record:       FLOW-RECORD-1
  Flow Exporter:     EXPORTER-1
  Cache:
    Type:              normal
    Status:            allocated

    Inactive Timeout:  15 secs
    Active Timeout:    1800 secs
    Update Timeout:    1800 secs
```

Displaying the Data in the Flow Monitor Cache

Perform this optional task to display the data in the flow monitor cache.

Before you begin

The interface on which you applied the input flow monitor must be receiving traffic that meets the criteria defined by the NetFlow original record before you can display the flows in the flow monitor cache.

SUMMARY STEPS

1. **enable**
2. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

Step 1 **enable**

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 **show flow monitor name** *monitor-name* **cache format record**

The **show flow monitor name** *monitor-name* **cache format record** command string displays the status, statistics, and flow data in the cache for a flow monitor.

Example:

```
Device# show flow monitor name FLOW-MONITOR-1 cache format record

Cache type:                               Normal

  Current entries:                           4
  High Watermark:                            4
  Flows added:                             101
  Flows aged:                               97
    - Active timeout    (  1800 secs)        3
    - Inactive timeout (    15 secs)        94
    - Event aged                             0
    - Watermark aged                         0
    - Emergency aged                         0
IPV4 DESTINATION ADDRESS:  172.16.10.5
ipv4 source address:       10.10.11.1
trns source port:          25
trns destination port:     25
counter bytes:             72840
counter packets:           1821
IPV4 DESTINATION ADDRESS:  172.16.10.2
ipv4 source address:       10.10.10.2
trns source port:          20
trns destination port:     20
counter bytes:             3913860
counter packets:           7326
IPV4 DESTINATION ADDRESS:  172.16.10.200
ipv4 source address:       192.168.67.6
trns source port:          0
trns destination port:     3073
counter bytes:             51072
counter packets:           1824
```

```
Device# show flow monitor name FLOW-MONITOR-2 cache format record

Cache type:                               Normal

  Current entries:                           2
  High Watermark:                            3
  Flows added:                              95
  Flows aged:                               93
    - Active timeout   (  1800 secs)         0
    - Inactive timeout (    15 secs)        93
    - Event aged                             0
    - Watermark aged                         0
    - Emergency aged                         0
IPV6 DESTINATION ADDRESS:  2001:DB8:4:ABCD::2
ipv6 source address:       2001:DB8:1:ABCD::1
trns source port:          33572
trns destination port:     23
counter bytes:             19140
counter packets:           349
IPV6 DESTINATION ADDRESS:  FF02::9
ipv6 source address:       FE80::A8AA:BBFF:FEBB:CC03
trns source port:          521
trns destination port:     521
counter bytes:             92
counter packets:           1
```

Verifying the Flow Monitor Configuration

Perform this optional task to verify the configuration commands that you entered.

SUMMARY STEPS

1. **enable**
2. **show running-config flow monitor**

DETAILED STEPS

Step 1 enable

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 show running-config flow monitor

The **show running-config flow monitor** command shows the configuration commands of the flow monitor that you specify.

Example:

```
Device# show running-config flow monitor FLOW-MONITOR-1
```

```
Current configuration:
!
flow monitor FLOW-MONITOR-1
 description Used for basic ipv4 traffic analysis
 record FLOW-RECORD-1
 exporter EXPORTER-1

!
```

Applying a Flow Monitor to an Interface

Before it can be activated, a flow monitor must be applied to at least one interface. Perform this required task to activate a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
5. Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.
6. **end**
7. **show flow interface** *type number*
8. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 4	{**ip**	**ipv6**} **flow monitor** *monitor-name* {**input**	**output**} **Example:**	Activates a flow monitor that was created previously by assigning it to the interface to analyze traffic.

	Command or Action	Purpose
	`Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	
Step 5	Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.	—
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show flow interface** *type number* **Example:** `Device# show flow interface GigabitEthernet 0/0/0`	Displays the status of Flexible NetFlow (enabled or disabled) on the specified interface.
Step 8	**show flow monitor name** *monitor-name* **cache format record** **Example:** `Device# show flow monitor name FLOW_MONITOR-1 cache format record`	Displays the status, statistics, and flow data in the cache for the specified flow monitor.

Verifying That Flexible NetFlow Is Enabled on an Interface

Perform this optional task to verify that Flexible NetFlow is enabled on an interface.

SUMMARY STEPS

1. **enable**
2. **show flow interface** *type number*

DETAILED STEPS

Step 1 **enable**

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 **show flow interface** *type number*

The **show flow interface** command verifies that Flexible NetFlow is enabled on an interface.

Example:

```
Device# show flow interface GigabitEthernet 0/0/0

Interface GigabitEthernet0/0/0
  FNF:  monitor:          FLOW-MONITOR-1
        direction:        Input
        traffic(ip):      on
  FNF:  monitor:          FLOW-MONITOR-2
        direction:        Input
        traffic(ipv6):    on
Device# show flow interface GigabitEthernet 1/0/0
Interface GigabitEthernet1/0/0
  FNF:  monitor:          FLOW-MONITOR-1
        direction:        Output
        traffic(ip):      on
  FNF:  monitor:          FLOW-MONITOR-2
        direction:        Input
        traffic(ipv6):    on
```

Configuration Examples for Flexible Netflow

Example: Configuring a Flexible NetFlow Predefined Record for IPv4 Traffic

The following example shows how to configure a flow monitor using the Flexible NetFlow "BGP ToS next-hop" predefined record to monitor IPv4 traffic.

This sample starts in global configuration mode:

```
!
flow monitor FLOW-MONITOR-1
 record netflow ipv4 bgp-nexthop-tos
 exit
!
ip cef
!
interface Ethernet 0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input
!
```

Example: Configuring a Flexible NetFlow Predefined Record for IPv6 Traffic

The following example shows how to configure a flow monitor using the Flexible NetFlow "source prefix" predefined record to monitor IPv6 traffic.

This sample starts in global configuration mode:

```
!
flow monitor FLOW-MONITOR-2
 record netflow ipv6 source-prefix
 exit
ip cef
ipv6 cef
!
```

```
interface GigabitEthernet 0/0/0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ipv6 flow monitor FLOW-MONITOR-2 input
!
```

Example: Configuring a Normal Flow Record Cache with a Limited Number of Flows

The following example is designed to monitor the type of service (ToS) field usage on all interfaces in the router. An exporter is not configured because this example is intended to be used to capture additional data for analysis on the router using the **show flow monitor** command.

This example starts in global configuration mode.

```
!
flow record QOS_RECORD
 description UD: Flow Record to monitor the use of TOS within this router/network
 match interface input
 match interface output
 match ipv4 tos
 collect counter packets
 collect counter bytes
 exit
!
flow monitor QOS_MONITOR
 description UD: Flow Monitor which watches the limited combinations of interface and TOS
 record QOS_RECORD
 cache type normal
 cache entries 8192   ! 2^5 (combos of interfaces) * 256 (values of TOS)
 exit
!
interface GigabitEthernet0/0/0
 ip flow monitor QOS_MONITOR input
 exit
!
interface GigabitEthernet0/1/0
 ip flow monitor QOS_MONITOR input
 exit
!
interface GigabitEthernet0/2/0
 ip flow monitor QOS_MONITOR input
 exit
!
```

The display from the **show flow monitor** command shows the current status of the cache.

```
Router# show flow monitor QOS_MONITOR cache

  Cache type:            Normal
  Cache size:                              8192
  Current entries:                            2
  High Watermark:                             2
  Flows added:                                2
  Updates sent         (  1800 secs)          0
```

Example: Configuring a Customized Flow Record Cache for Monitoring IPv6 Traffic

The following example creates a customized flow record cache for monitoring IPv6 traffic.

This example starts in global configuration mode.

Example: Configuring a Permanent Flow Record Cache with a Limited Number of Flows

The following example is designed to monitor the type of service (ToS) field usage on all interfaces in the router. An exporter is not configured because this example is intended to be used to capture additional data for analysis on the router using the **show flow monitor**command.

This example starts in global configuration mode.

```
!
ip cef
!
flow record QOS_RECORD
 description UD: Flow Record to monitor the use of TOS within this router/network
 match interface input
 match interface output
 match ipv4 tos
 collect counter packets
 collect counter bytes
 exit
!
flow monitor QOS_MONITOR
 description UD: Flow Monitor which watches the limited combinations of interface and TOS
 record QOS_RECORD
 cache type permanent
 cache entries 8192   ! 2^5 (combos of interfaces) * 256 (values of TOS)
 exit
!
interface ethernet0/0
 ip flow monitor QOS_MONITOR input
 exit
!
interface ethernet0/1
 ip flow monitor QOS_MONITOR input
 exit
!
interface ethernet0/2
 ip flow monitor QOS_MONITOR input
 exit
!
interface serial2/0
 ip flow monitor QOS_MONITOR input
 exit
!
interface serial2/1
 ip flow monitor QOS_MONITOR input
!
```

The display from the **show flow monitor** command shows the current status of the cache.

```
Router# show flow monitor QOS_MONITOR cache
```

```
  Cache type:                           Permanent
  Cache size:                                8192
  Current entries:                              2
  High Watermark:                               2
  Flows added:                                  2
  Updates sent         (  1800 secs)            0
```

Example: Configuring Flexible NetFlow Egress Accounting for IPv4 and IPv6 Traffic

The following example shows how to configure Flexible NetFlow egress accounting for IPv4 and IPv6 traffic.

This example starts in global configuration mode.

```
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow record v6_r1
match ipv6 traffic-class
match ipv6 protocol
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exit
!
!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exit
!
ip cef
ipv6 cef
!
interface GigabitEthernet0/0/0
 ip address 172.16.6.2 255.255.255.0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ip flow monitor FLOW-MONITOR-1 output
 ipv6 flow monitor FLOW-MONITOR-2 output
!
```

Example: Configuring Flexible NetFlow Subinterface Support

The following example shows how to configure Flexible NetFlow subinterface support for IPv4 traffic.

This example starts in global configuration mode.

```
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exit
!
ip cef
!
interface Ethernet0/0.1
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input
!
```

The following example shows how to configure Flexible NetFlow to emulate NetFlow subinterface support for IPv6 traffic.

This sample starts in global configuration mode:

```
!
flow record v6_r1

match ipv6 protocol
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long

!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exit
!
ip cef
ipv6 cef
!
interface Ethernet0/0.1
 ipv6 address 2001:DB8:2:ABCD::2/48
 ipv6 flow monitor FLOW-MONITOR-2 input
!
```

Example: Configuring Flexible NetFlow Multiple Export Destinations

The following example shows how to configure Flexible NetFlow multiple export destinations.

This example starts in global configuration mode.

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
```

```
 transport udp 90
 exit
!
flow exporter EXPORTER-2
 destination 172.16.10.3
 transport udp 90
 exit
!
flow monitor FLOW-MONITOR-1
 record netflow-original
 exporter EXPORTER-2
 exporter EXPORTER-1
 exit
!
ip cef
!
interface GigabitEthernet0/0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input
!
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

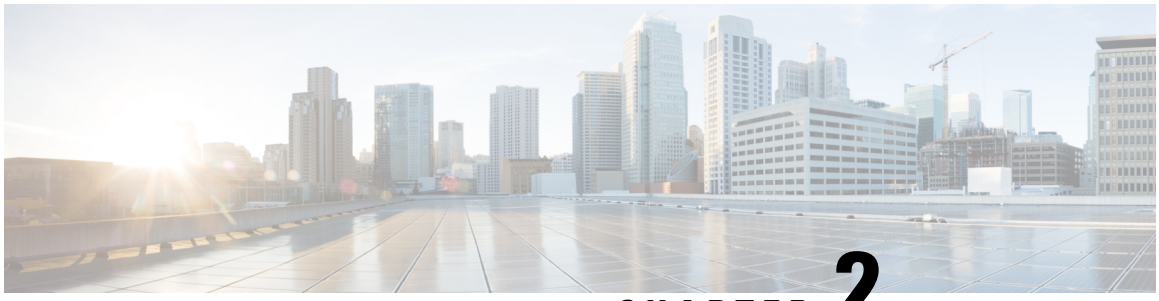
Feature Information for Flexible NetFlow

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

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

Table 19: Feature Information for Flexible NetFlow

Feature Name	Releases	Feature Information
Flexible NetFlow	12.2(33)SRC 12.2(50)SY 12.4(9)T 15.0(1)SY 15.0(1)SY1 Cisco IOS XE Release 3.1S	Flexible NetFlow is introduced. Support for this feature was added for Cisco 7200 series routers in Cisco IOS Release 12.2(33)SRC. The following commands were introduced or modified: **cache** (Flexible NetFlow), **clear flow exporter**, **clear flow monitor**, **clear sampler**, **collect counter**, **collect flow**, **collect interface**, **collect ipv4**, **collect ipv4 destination**, **collect ipv4 fragmentation**, **collect ipv4 section**, **collect ipv4 source**, **collect ipv4 total-length**, **collect ipv4 ttl**, **collect routing**, **collect timestamp sys-uptime**, **collect transport**, **collect transport icmp ipv4**, **collect transport tcp**, **collect transport udp**, **debug flow exporter**, **debug flow monitor**, **debug flow record**, **debug sampler**, **description** (Flexible NetFlow), **destination**, **dscp** (Flexible NetFlow), **exporter**, **flow exporter**, **flow monitor**, **flow platform**, **flow record**, **ip flow monitor**, **match flow**, **match interface** (Flexible NetFlow), **match ipv4**, **match ipv4 destination**, **match ipv4 fragmentation**, **match ipv4 section**, **match ipv4 source**, **match ipv4 total-length**, **match ipv4 ttl**, **match routing**, **match transport**, **match transport icmp ipv4**, **match transport tcp**, **match transport udp**, **mode** (Flexible NetFlow), **option** (Flexible NetFlow), **record**, **sampler**, **show flow exporter**, **show flow interface**, **show flow monitor**, **show flow record**, **show sampler**, **source** (Flexible NetFlow), **statistics packet**, **template data timeout**, **transport** (Flexible NetFlow).



CHAPTER 2

Flexible NetFlow—IPv4 Unicast Flows

The Flexible Netflow—IPv4 Unicast Flows feature enables Flexible NetFlow to monitor IPv4 traffic.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible NetFlow IPv4 Unicast Flows

Flexible NetFlow—IPv4 Unicast Flows Overview

This feature enables Flexible NetFlow to monitor IPv4 traffic.

How to Configure Flexible NetFlow IPv4 Unicast Flows

Creating a Customized Flow Record

Perform this task to configure a customized flow record.

Customized flow records are used to analyze traffic data for a specific purpose. A customized flow record must have at least one **match** criterion for use as the key field and typically has at least one **collect** criterion for use as a nonkey field.

There are hundreds of possible permutations of customized flow records. This task shows the steps that are used to create one of the possible permutations. Modify the steps in this task as appropriate to create a customized flow record for your requirements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **description** *description*
5. **match** {**ip** | **ipv6**} {**destination** | **source**} **address**
6. Repeat Step 5 as required to configure additional key fields for the record.
7. **collect interface** {**input** | **output**}
8. Repeat the above step as required to configure additional nonkey fields for the record.
9. **end**
10. **show flow record** *record-name*
11. **show running-config flow record** *record-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-RECORD-1`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. • This command also allows you to modify an existing flow record.
Step 4	**description** *description* **Example:** `Device(config-flow-record)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow record.
Step 5	**match** {**ip**	**ipv6**} {**destination**	**source**} **address** **Example:**	Configures a key field for the flow record.

	Command or Action	Purpose
	`Device(config-flow-record)# match ipv4 destination address`	**Note** This example configures the IPv4 destination address as a key field for the record. For information about the other key fields available for the **match ipv4** command, and the other **match** commands that are available to configure key fields.
Step 6	Repeat Step 5 as required to configure additional key fields for the record.	—
Step 7	**collect interface** {**input** | **output**} **Example:**	Configures the input interface as a nonkey field for the record. **Note** This example configures the input interface as a nonkey field for the record.
Step 8	Repeat the above step as required to configure additional nonkey fields for the record.	—
Step 9	**end** **Example:** `Device(config-flow-record)# end`	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 10	**show flow record** *record-name* **Example:** `Device# show flow record FLOW_RECORD-1`	(Optional) Displays the current status of the specified flow record.
Step 11	**show running-config flow record** *record-name* **Example:** `Device# show running-config flow record FLOW_RECORD-1`	(Optional) Displays the configuration of the specified flow record.

Configuring the Flow Exporter

Perform this required task to configure the flow exporter.

Note Each flow exporter supports only one destination.

You can export to a destination using either an IPv4 or IPv6 address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **dscp** *dscp*
7. **source** *interface-type interface-number*
8. **output-features**
9. **template data timeout** *seconds*
10. **transport udp** *udp-port*
11. **ttl** *seconds*
12. **end**
13. **show flow exporter** *exporter-name*
14. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to the datacenter`	(Optional) Configures a description to the exporter that will appear in the configuration and the display of the **show flow exporter** command.
Step 5	**destination** {*ip-address*	*hostname*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the IP address or hostname of the destination system for the exporter. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**dscp** *dscp* **Example:** `Device(config-flow-exporter)# dscp 63`	(Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter. • The range for the *dscp* argument is from 0 to 63. Default: 0.

	Command or Action	Purpose
Step 7	**source** *interface-type interface-number* **Example:** `Device(config-flow-exporter)# source ethernet 0/0`	(Optional) Specifies the local interface from which the exporter will use the IP address as the source IP address for exported datagrams.
Step 8	**output-features** **Example:** `Device(config-flow-exporter)# output-features`	(Optional) Enables sending export packets using quality of service (QoS) and encryption.
Step 9	**template data timeout** *seconds* **Example:** `Device(config-flow-exporter)# template data timeout 120`	(Optional) Configures resending of templates based on a timeout. • The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours).
Step 10	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 650`	Specifies the UDP port on which the destination system is listening for exported datagrams. • The range for the *udp-port* argument is from 1 to 65536.
Step 11	**ttl** *seconds* **Example:** `Device(config-flow-exporter)# ttl 15`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. • The range for the *seconds* argument is from 1 to 255.
Step 12	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flow exporter configuration mode and returns to privileged EXEC mode.
Step 13	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 14	**show running-config flow exporter** *exporter-name* **Example:** `Device# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Creating a Customized Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be one of the predefined formats or a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record instead of using one of the Flexible NetFlow predefined records, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}}
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **statistics packet protocol**
9. **statistics packet size**
10. **exporter** *exporter-name*
11. **end**
12. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]]
13. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:**	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode.

	Command or Action	Purpose
	`(config)# flow monitor FLOW-MONITOR-1`	• This command also allows you to modify an existing flow monitor.
Step 4	**description** *description* **Example:** `(config-flow-monitor)# description Used for basic ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}} **Example:**	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. The values for the keywords associated with the **timeout** keyword have no effect when the cache type is set to **immediate**.
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	**statistics packet protocol** **Example:** `(config-flow-monitor)# statistics packet protocol`	(Optional) Enables the collection of protocol distribution statistics for Flexible NetFlow monitors.
Step 9	**statistics packet size** **Example:** `(config-flow-monitor)# statistics packet size`	(Optional) Enables the collection of size distribution statistics for Flexible NetFlow monitors.
Step 10	**exporter** *exporter-name* **Example:** `(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.
Step 11	**end** **Example:** `(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] **Example:**	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.

	Command or Action	Purpose
	`# show flow monitor FLOW-MONITOR-2 cache`	
Step 13	**show running-config flow monitor** *monitor-name* **Example:** `# show running-config flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Applying a Flow Monitor to an Interface

Before it can be activated, a flow monitor must be applied to at least one interface. Perform this required task to activate a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
5. Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.
6. **end**
7. **show flow interface** *type number*
8. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 4	{**ip**	**ipv6**} **flow monitor** *monitor-name* {**input**	**output**} **Example:**	Activates a flow monitor that was created previously by assigning it to the interface to analyze traffic.

	Command or Action	Purpose
	`Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	
Step 5	Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.	—
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show flow interface** *type number* **Example:** `Device# show flow interface GigabitEthernet 0/0/0`	Displays the status of Flexible NetFlow (enabled or disabled) on the specified interface.
Step 8	**show flow monitor name** *monitor-name* **cache format record** **Example:** `Device# show flow monitor name FLOW_MONITOR-1 cache format record`	Displays the status, statistics, and flow data in the cache for the specified flow monitor.

Configuring and Enabling Flexible NetFlow with Data Export

You must create a flow monitor to configure the types of traffic for which you want to export the cache data. You must enable the flow monitor by applying it to at least one interface to start exporting data. To configure and enable Flexible NetFlow with data export, perform this required task.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. The record format can be one of the predefined record formats, or an advanced user may create his or her own record format using the **collect** and **match**commands in Flexible NetFlow flow record configuration mode.

Note You must remove a flow monitor from all of the interfaces to which you have applied it before you can modify the **record** format of the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6** *record* [**peer**] }]
5. **exporter** *exporter-name*
6. **exit**

7. **interface** *type number*
8. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
9. **end**
10. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.
Step 4	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6** *record* [**peer**] }] **Example:** `Device(config-flow-monitor)# record netflow ipv4 original-input`	Specifies the record for the flow monitor.
Step 5	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter EXPORTER-1`	Specifies the name of an exporter that you created previously.
Step 6	**exit** **Example:** `Device(config-flow-monitor)# exit`	Exits Flexible NetFlow flow monitor configuration mode and returns to global configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 8	{**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**} **Example:**	Activates the flow monitor that you created previously by assigning it to the interface to analyze traffic.

	Command or Action	Purpose
	`Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	
Step 9	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 10	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]] **Example:** `Device# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor. This will verify data export is enabled for the flow monitor cache.

Configuration Examples for Flexible NetFlow IPv4 Unicast Flows

Example: Configuring Multiple Export Destinations

The following example shows how to configure multiple export destinations for Flexible NetFlow for IPv4 or IPv6 traffic.

This sample starts in global configuration mode:

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
 transport udp 90
 exit
!
flow exporter EXPORTER-2
 destination 172.16.10.3
 transport udp 90
 exit
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow record v6_r1
match ipv6 traffic-class
match ipv6 protocol
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
```

```
collect counter packets long
!

flow monitor FLOW-MONITOR-1
 record v4_r1
 exporter EXPORTER-2
 exporter EXPORTER-1
!
!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exporter EXPORTER-2
 exporter EXPORTER-1
!
ip cef
!
interface GigabitEthernet1/0/0
 ip address 172.16.6.2 255.255.255.0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ip flow monitor FLOW-MONITOR-1 input
 ipv6 flow monitor FLOW-MONITOR-2 input
!
```

The following display output shows that the flow monitor is exporting data to the two exporters:

```
Device# show flow monitor FLOW-MONITOR-1
Flow Monitor FLOW-MONITOR-1:
  Description:       User defined
  Flow Record:       v4_r1
  Flow Exporter:     EXPORTER-1
                     EXPORTER-2
  Cache:
    Type:              normal (Platform cache)
    Status:            allocated
    Size:              4096 entries / 311316 bytes
    Inactive Timeout:  15 secs
    Active Timeout:    1800 secs
    Update Timeout:    1800 secs
```

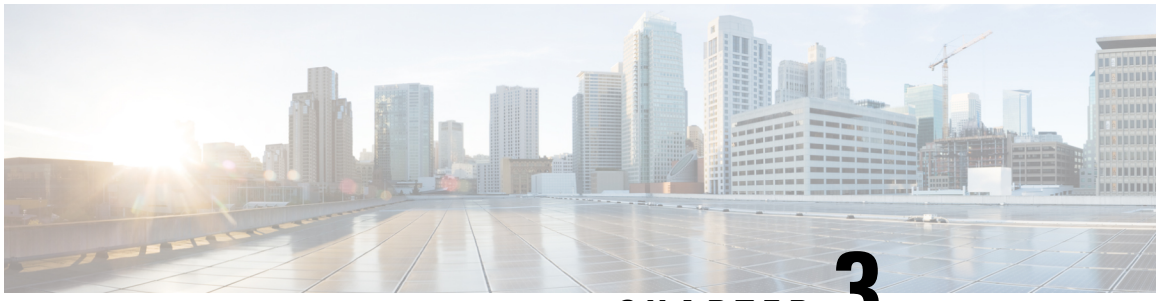
Example: Configuring Flexible NetFlow Egress Accounting for IPv4 and IPv6 Traffic

The following example shows how to configure Flexible NetFlow egress accounting for IPv4 and IPv6 traffic.

This example starts in global configuration mode.

```
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow record v6_r1
match ipv6 traffic-class
match ipv6 protocol
```

```
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exit
!
!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exit
!
ip cef
ipv6 cef
!
interface GigabitEthernet0/0/0
 ip address 172.16.6.2 255.255.255.0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ip flow monitor FLOW-MONITOR-1 output
 ipv6 flow monitor FLOW-MONITOR-2 output
!
```



CHAPTER 3

Flexible NetFlow—IPv6 Unicast Flows

The Flexible NetFlow—IPv6 Unicast Flows feature enables Flexible NetFlow to monitor IPv6 traffic.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible NetFlow IPv6 Unicast Flows

Flexible NetFlow IPv6 Unicast Flows Overview

This feature enables Flexible NetFlow to monitor IPv6 traffic.

How to Configure Flexible NetFlow IPv6 Unicast Flows

Creating a Customized Flow Record

Perform this task to configure a customized flow record.

Customized flow records are used to analyze traffic data for a specific purpose. A customized flow record must have at least one **match** criterion for use as the key field and typically has at least one **collect** criterion for use as a nonkey field.

There are hundreds of possible permutations of customized flow records. This task shows the steps that are used to create one of the possible permutations. Modify the steps in this task as appropriate to create a customized flow record for your requirements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **description** *description*
5. **match** {**ip** | **ipv6**} {**destination** | **source**} **address**
6. Repeat Step 5 as required to configure additional key fields for the record.
7. **collect interface** {**input** | **output**}
8. Repeat the above step as required to configure additional nonkey fields for the record.
9. **end**
10. **show flow record** *record-name*
11. **show running-config flow record** *record-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-RECORD-1`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. • This command also allows you to modify an existing flow record.
Step 4	**description** *description* **Example:** `Device(config-flow-record)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow record.
Step 5	**match** {**ip**	**ipv6**} {**destination**	**source**} **address** **Example:**	Configures a key field for the flow record.

	Command or Action	Purpose
	`Device(config-flow-record)# match ipv4 destination address`	**Note** This example configures the IPv4 destination address as a key field for the record. For information about the other key fields available for the **match ipv4** command, and the other **match** commands that are available to configure key fields.
Step 6	Repeat Step 5 as required to configure additional key fields for the record.	—
Step 7	**collect interface** {**input** | **output**} **Example:**	Configures the input interface as a nonkey field for the record. **Note** This example configures the input interface as a nonkey field for the record.
Step 8	Repeat the above step as required to configure additional nonkey fields for the record.	—
Step 9	**end** **Example:** `Device(config-flow-record)# end`	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 10	**show flow record** *record-name* **Example:** `Device# show flow record FLOW_RECORD-1`	(Optional) Displays the current status of the specified flow record.
Step 11	**show running-config flow record** *record-name* **Example:** `Device# show running-config flow record FLOW_RECORD-1`	(Optional) Displays the configuration of the specified flow record.

Configuring the Flow Exporter

Perform this required task to configure the flow exporter.

Note Each flow exporter supports only one destination.

You can export to a destination using either an IPv4 or IPv6 address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **dscp** *dscp*
7. **source** *interface-type interface-number*
8. **output-features**
9. **template data timeout** *seconds*
10. **transport udp** *udp-port*
11. **ttl** *seconds*
12. **end**
13. **show flow exporter** *exporter-name*
14. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to the datacenter`	(Optional) Configures a description to the exporter that will appear in the configuration and the display of the **show flow exporter** command.
Step 5	**destination** {*ip-address*	*hostname*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the IP address or hostname of the destination system for the exporter. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**dscp** *dscp* **Example:** `Device(config-flow-exporter)# dscp 63`	(Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter. • The range for the *dscp* argument is from 0 to 63. Default: 0.

	Command or Action	Purpose
Step 7	**source** *interface-type interface-number* **Example:** `Device(config-flow-exporter)# source ethernet 0/0`	(Optional) Specifies the local interface from which the exporter will use the IP address as the source IP address for exported datagrams.
Step 8	**output-features** **Example:** `Device(config-flow-exporter)# output-features`	(Optional) Enables sending export packets using quality of service (QoS) and encryption.
Step 9	**template data timeout** *seconds* **Example:** `Device(config-flow-exporter)# template data timeout 120`	(Optional) Configures resending of templates based on a timeout. • The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours).
Step 10	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 650`	Specifies the UDP port on which the destination system is listening for exported datagrams. • The range for the *udp-port* argument is from 1 to 65536.
Step 11	**ttl** *seconds* **Example:** `Device(config-flow-exporter)# ttl 15`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. • The range for the *seconds* argument is from 1 to 255.
Step 12	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flow exporter configuration mode and returns to privileged EXEC mode.
Step 13	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 14	**show running-config flow exporter** *exporter-name* **Example:** `Device# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Creating a Customized Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be one of the predefined formats or a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record instead of using one of the Flexible NetFlow predefined records, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}}
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **statistics packet protocol**
9. **statistics packet size**
10. **exporter** *exporter-name*
11. **end**
12. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]]
13. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:**	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode.

	Command or Action	Purpose
	`(config)# flow monitor FLOW-MONITOR-1`	• This command also allows you to modify an existing flow monitor.
Step 4	**description** *description* **Example:** `(config-flow-monitor)# description Used for basic ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}} **Example:**	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. The values for the keywords associated with the **timeout** keyword have no effect when the cache type is set to **immediate**.
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	**statistics packet protocol** **Example:** `(config-flow-monitor)# statistics packet protocol`	(Optional) Enables the collection of protocol distribution statistics for Flexible NetFlow monitors.
Step 9	**statistics packet size** **Example:** `(config-flow-monitor)# statistics packet size`	(Optional) Enables the collection of size distribution statistics for Flexible NetFlow monitors.
Step 10	**exporter** *exporter-name* **Example:** `(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.
Step 11	**end** **Example:** `(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] **Example:**	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.

	Command or Action	Purpose
	`# show flow monitor FLOW-MONITOR-2 cache`	
Step 13	**show running-config flow monitor** *monitor-name* **Example:** `# show running-config flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Applying a Flow Monitor to an Interface

Before it can be activated, a flow monitor must be applied to at least one interface. Perform this required task to activate a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
5. Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.
6. **end**
7. **show flow interface** *type number*
8. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 4	{**ip**	**ipv6**} **flow monitor** *monitor-name* {**input**	**output**} **Example:**	Activates a flow monitor that was created previously by assigning it to the interface to analyze traffic.

	Command or Action	Purpose
	`Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	
Step 5	Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.	—
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show flow interface** *type number* **Example:** `Device# show flow interface GigabitEthernet 0/0/0`	Displays the status of Flexible NetFlow (enabled or disabled) on the specified interface.
Step 8	**show flow monitor name** *monitor-name* **cache format record** **Example:** `Device# show flow monitor name FLOW_MONITOR-1 cache format record`	Displays the status, statistics, and flow data in the cache for the specified flow monitor.

Configuring and Enabling Flexible NetFlow with Data Export

You must create a flow monitor to configure the types of traffic for which you want to export the cache data. You must enable the flow monitor by applying it to at least one interface to start exporting data. To configure and enable Flexible NetFlow with data export, perform this required task.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. The record format can be one of the predefined record formats, or an advanced user may create his or her own record format using the **collect** and **match**commands in Flexible NetFlow flow record configuration mode.

Note You must remove a flow monitor from all of the interfaces to which you have applied it before you can modify the **record** format of the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6** *record* [**peer**] }]
5. **exporter** *exporter-name*
6. **exit**

7. **interface** *type number*
8. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
9. **end**
10. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.
Step 4	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6** *record* [**peer**] }] **Example:** `Device(config-flow-monitor)# record netflow ipv4 original-input`	Specifies the record for the flow monitor.
Step 5	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter EXPORTER-1`	Specifies the name of an exporter that you created previously.
Step 6	**exit** **Example:** `Device(config-flow-monitor)# exit`	Exits Flexible NetFlow flow monitor configuration mode and returns to global configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 8	{**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**} **Example:**	Activates the flow monitor that you created previously by assigning it to the interface to analyze traffic.

	Command or Action	Purpose
	`Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	
Step 9	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 10	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]] **Example:** `Device# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor. This will verify data export is enabled for the flow monitor cache.

Configuration Examples for Flexible NetFlow IPv6 Unicast Flows

Example: Configuring Multiple Export Destinations

The following example shows how to configure multiple export destinations for Flexible NetFlow for IPv4 or IPv6 traffic.

This sample starts in global configuration mode:

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
 transport udp 90
 exit
!
flow exporter EXPORTER-2
 destination 172.16.10.3
 transport udp 90
 exit
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow record v6_r1
match ipv6 traffic-class
match ipv6 protocol
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
```

```
collect counter packets long
!

flow monitor FLOW-MONITOR-1
 record v4_r1
 exporter EXPORTER-2
 exporter EXPORTER-1
!
!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exporter EXPORTER-2
 exporter EXPORTER-1
!
ip cef
!
interface GigabitEthernet1/0/0
 ip address 172.16.6.2 255.255.255.0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ip flow monitor FLOW-MONITOR-1 input
 ipv6 flow monitor FLOW-MONITOR-2 input
!
```

The following display output shows that the flow monitor is exporting data to the two exporters:

```
Device# show flow monitor FLOW-MONITOR-1
Flow Monitor FLOW-MONITOR-1:
  Description:       User defined
  Flow Record:       v4_r1
  Flow Exporter:     EXPORTER-1
                     EXPORTER-2
  Cache:
    Type:              normal (Platform cache)
    Status:            allocated
    Size:              4096 entries / 311316 bytes
    Inactive Timeout:  15 secs
    Active Timeout:    1800 secs
    Update Timeout:    1800 secs
```

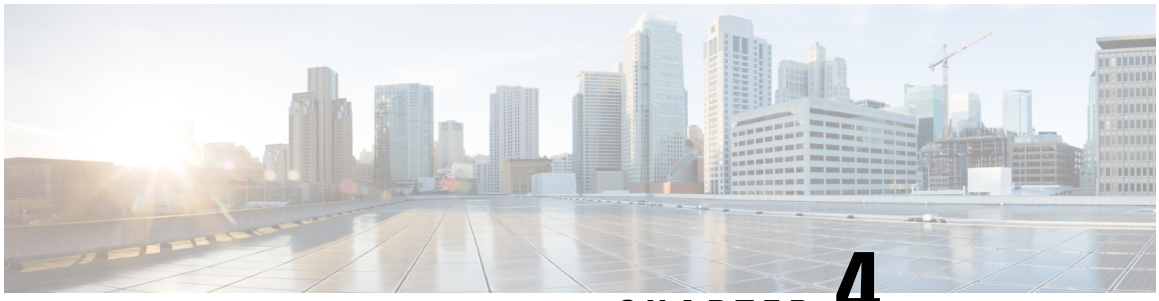
Example: Configuring Flexible NetFlow Egress Accounting for IPv4 and IPv6 Traffic

The following example shows how to configure Flexible NetFlow egress accounting for IPv4 and IPv6 traffic.

This example starts in global configuration mode.

```
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow record v6_r1
match ipv6 traffic-class
match ipv6 protocol
```

```
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exit
!
!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exit
!
ip cef
ipv6 cef
!
interface GigabitEthernet0/0/0
 ip address 172.16.6.2 255.255.255.0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ip flow monitor FLOW-MONITOR-1 output
 ipv6 flow monitor FLOW-MONITOR-2 output
!
```



CHAPTER 4

Flexible NetFlow—MPLS Egress NetFlow

The Flexible NetFlow--MPLS Egress NetFlow feature allows you to capture IP flow information for packets undergoing MPLS label disposition; that is, packets that arrive on a router as MPLS packets and are transmitted as IP packets.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible NetFlow MPLS Egress NetFlow

Flexible NetFlow MPLS Egress NetFlow

The Flexible NetFlow - MPLS Egress NetFlow feature allows you to capture IP flow information for packets that arrive on a router as Multiprotocol Label Switching (MPLS) packets and are transmitted as IP packets. This feature allows you to capture the MPLS VPN IP flows that are traveling through the service provider backbone from one site of a VPN to another site of the same VPN. The Flexible NetFlow - MPLS Egress NetFlow feature is enabled by applying a flow monitor in output (egress) mode on the provider edge (PE) to customer edge (CE) interface of the provider's network.

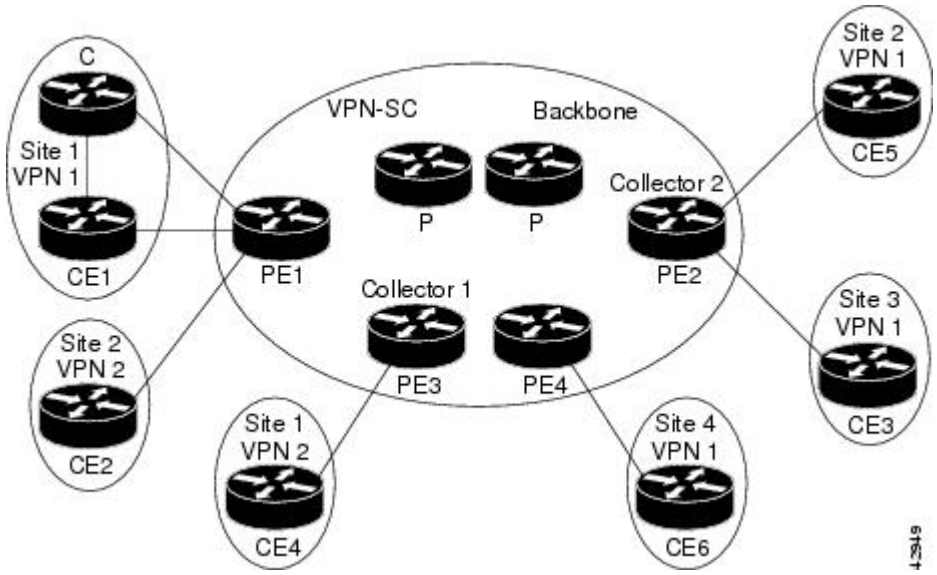
The figure below shows a sample MPLS VPN network topology that includes four VPN 1 sites and two VPN 2 sites. If the Flexible NetFlow - MPLS Egress NetFlow is enabled on an outgoing PE interface by applying

a flow monitor in output mode, IP flow information for packets that arrive at the PE as MPLS packets (from an MPLS VPN) and that are transmitted as IP packets to the PE router is captured. For example:

- To capture the flow of traffic going to site 2 of VPN 1 from any remote VPN 1 sites, you enable a flow monitor in output mode on link PE2-CE5 of provider edge router PE2.
- To capture the flow of traffic going to site 1 of VPN 2 from any remote VPN 2 site, you enable a flow monitor in output mode on link PE3-CE4 of the provider edge router PE3.

The flow data is stored in the Flexible NetFlow cache. You can use the **show flow monitor** *monitor-name* **cache**command to display the flow data in the cache.

Figure 7: Sample MPLS VPN Network Topology with Flexible NetFlow--MPLS Egress NetFlow Feature



If you configure a Flexible NetFlow exporter for the flow monitors you use for the Flexible NetFlow - MPLS Egress NetFlow feature, the PE routers will export the captured flows to the configured collector devices in the provider network. Applications such as the Network Data Analyzer or the VPN Solution Center (VPN-SC) can gather information from the captured flows and compute and display site-to-site VPN traffic statistics.

Limitations

When using Flexible NetFlow to monitor outbound traffic on a router at the edge of an MPLS cloud, for IP traffic that leaves over a VRF, the following fields are not collected and have a value of 0:

- destination mask
- destination prefix
- destination AS numbers
- destination BGP traffic index
- nexthop
- BGP nexthop

How to Configure Flexible NetFlow MPLS Egress NetFlow

Configuring a Flow Exporter for the Flow Monitor

Perform this optional task to configure a flow exporter for the flow monitor in order to export the data that is collected by Flexible NetFlow to a remote system for further analysis and storage.

Flow exporters are used to send the data that you collect with Flexible NetFlow to a remote system such as a NetFlow Collection Engine. Exporters use UDP as the transport protocol and use the Version 9 export format.

Note Each flow exporter supports only one destination. If you want to export the data to multiple destinations, you must configure multiple flow exporters and assign them to the flow monitor.

You can export to a destination using either an IPv4 or IPv6 address.

Note When you configure an exporter, configure the exporter in such a way that the source interface is defined as a WAN interface. This configuration helps you prevent any unpredictable behavior because the NAT is not applied on the packets.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*hostname* | *ip-address*} [**vrf** *vrf-name*]
6. **export-protocol** {**netflow-v5** | **netflow-v9** | **ipfix**}
7. **transport udp** *udp-port*
8. **exit**
9. **flow monitor** *flow-monitor-name*
10. **exporter** *exporter-name*
11. **end**
12. **show flow exporter** *exporter-name*
13. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates a flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to datacenter`	(Optional) Creates a description for the flow exporter.
Step 5	**destination** {*hostname* | *ip-address*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the hostname or IP address of the system to which the exporter sends data. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**export-protocol** {**netflow-v5** | **netflow-v9** | **ipfix**} **Example:** `Device(config-flow-exporter)# export-protocol netflow-v9`	Specifies the version of the NetFlow export protocol used by the exporter. • Default: **netflow-v9**.
Step 7	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 65`	Configures UDP as the transport protocol and specifies the UDP port on which the destination system is listening for exported Flexible NetFlow traffic.
Step 8	**exit** **Example:** `Device(config-flow-exporter)# exit`	Exits Flexible NetFlow flow exporter configuration mode and returns to global configuration mode.
Step 9	**flow monitor** *flow-monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Enters Flexible NetFlow flow monitor configuration mode for the flow monitor that you created previously.
Step 10	**exporter** *exporter-name* **Example:**	Specifies the name of an exporter that you created previously.

	Command or Action	Purpose
	`Device(config-flow-monitor)# exporter EXPORTER-1`	
Step 11	**end** **Example:** `Device(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 13	**show running-config flow exporter** *exporter-name* **Example:** `Device<# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Creating a Customized Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be one of the predefined formats or a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record instead of using one of the Flexible NetFlow predefined records, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}}

7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **statistics packet protocol**
9. **statistics packet size**
10. **exporter** *exporter-name*
11. **end**
12. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]]
13. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.
Step 4	**description** *description* **Example:** `(config-flow-monitor)# description Used for basic ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}} **Example:**	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. The values for the keywords associated with the **timeout** keyword have no effect when the cache type is set to **immediate**.
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—

	Command or Action	Purpose
Step 8	**statistics packet protocol** **Example:** `(config-flow-monitor)# statistics packet protocol`	(Optional) Enables the collection of protocol distribution statistics for Flexible NetFlow monitors.
Step 9	**statistics packet size** **Example:** `(config-flow-monitor)# statistics packet size`	(Optional) Enables the collection of size distribution statistics for Flexible NetFlow monitors.
Step 10	**exporter** *exporter-name* **Example:** `(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.
Step 11	**end** **Example:** `(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] **Example:** `# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.
Step 13	**show running-config flow monitor** *monitor-name* **Example:** `# show running-config flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Applying a Flow Monitor to an Interface

Before it can be activated, a flow monitor must be applied to at least one interface. Perform this required task to activate a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
5. Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.
6. **end**
7. **show flow interface** *type number*

8. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 4	{**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**} **Example:** `Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	Activates a flow monitor that was created previously by assigning it to the interface to analyze traffic.
Step 5	Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.	—
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show flow interface** *type number* **Example:** `Device# show flow interface GigabitEthernet 0/0/0`	Displays the status of Flexible NetFlow (enabled or disabled) on the specified interface.
Step 8	**show flow monitor name** *monitor-name* **cache format record** **Example:** `Device# show flow monitor name FLOW_MONITOR-1 cache format record`	Displays the status, statistics, and flow data in the cache for the specified flow monitor.

Configuration Examples for Flexible NetFlow MPLS Egress NetFlow

Example: Configuring Flexible NetFlow Egress Accounting for IPv4 and IPv6 Traffic

The following example shows how to configure Flexible NetFlow egress accounting for IPv4 and IPv6 traffic.

This example starts in global configuration mode.

```
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow record v6_r1
match ipv6 traffic-class
match ipv6 protocol
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exit
!
!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exit
!
ip cef
ipv6 cef
!
interface GigabitEthernet0/0/0
 ip address 172.16.6.2 255.255.255.0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ip flow monitor FLOW-MONITOR-1 output
 ipv6 flow monitor FLOW-MONITOR-2 output
!
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

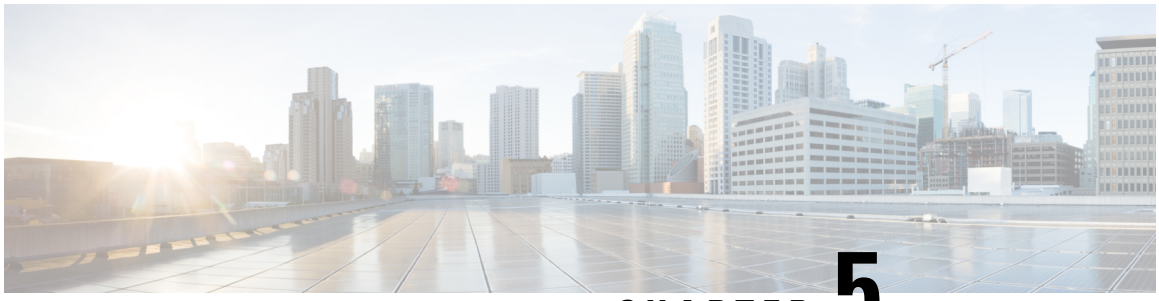
Feature Information for Flexible NetFlow - MPLS Egress NetFlow

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

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

Table 20: Feature Information for Flexible NetFlow - MPLS Egress NetFlow

Feature Name	Releases	Feature Information
Flexible NetFlow - MPLS Egress NetFlow	12.2(33)SRE 12.2(50)SY 12.4(22)T 15.0(1)SY 15.0(1)SY1 Cisco IOS XE Release 3.1S	The Flexible NetFlow--MPLS Egress NetFlow feature allows you to capture IP flow information for packets undergoing MPLS label disposition; that is, packets that arrive on a router as MPLS packets and are transmitted as IP packets. Support for this feature was added for Cisco 7200 and 7300 NPE series routers in Cisco IOS Release 12.2(33)SRE. No commands were introduced or modified by this feature.



CHAPTER 5

Flexible NetFlow v9 Export Format

This feature enables sending export packets using the Version 9 export format.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Prerequisites for Flexible NetFlow v9 Export Format

- The networking device must be running a Cisco release that supports Flexible NetFlow.

Information About Flexible NetFlow v9 Export Format

Flow Exporters

Flow exporters are created as separate components in a router's configuration. Exporters are assigned to flow monitors to export the data from the flow monitor cache to a remote system such as a NetFlow collector. Flow monitors can support more than one exporter. Each exporter can be customized to meet the requirements of the flow monitor or monitors in which it is used and the NetFlow collector systems to which it is exporting data.

Benefits of Flexible NetFlow Flow Exporters

Flexible NetFlow allows you to configure many different flow exporters, depending on your requirements. Some of the benefits of Flexible NetFlow flow exporters are as follows:

- Using flow exporters, you can create an exporter for every type of traffic that you want to analyze so that you can send each type of traffic to a different NetFlow collector. Original NetFlow sends the data in a cache for all of the analyzed traffic to a maximum of two export destinations.
- Flow exporters support up to ten exporters per flow monitor. Original NetFlow is limited to only two export destinations per cache.
- Flow exporters can use both TCP and UDP for export.
- Depending on your release, flow exporters can use class of service (CoS) in the packets that are sent to export destinations to help ensure that the packets are given the correct priority throughout the network. Original NetFlow exporters do not use CoS in the packets that are sent to export destinations.
- Depending on your release, flow exporter traffic can be encrypted.

How to Configure Flexible NetFlow v9 Export Format

Configuring the Flow Exporter

Perform this required task to configure the flow exporter.

Note Each flow exporter supports only one destination.

You can export to a destination using either an IPv4 or IPv6 address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **dscp** *dscp*
7. **source** *interface-type interface-number*
8. **output-features**
9. **template data timeout** *seconds*
10. **transport udp** *udp-port*
11. **ttl** *seconds*
12. **end**
13. **show flow exporter** *exporter-name*
14. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to the datacenter`	(Optional) Configures a description to the exporter that will appear in the configuration and the display of the **show flow exporter** command.
Step 5	**destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the IP address or hostname of the destination system for the exporter. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**dscp** *dscp* **Example:** `Device(config-flow-exporter)# dscp 63`	(Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter. • The range for the *dscp* argument is from 0 to 63. Default: 0.
Step 7	**source** *interface-type interface-number* **Example:** `Device(config-flow-exporter)# source ethernet 0/0`	(Optional) Specifies the local interface from which the exporter will use the IP address as the source IP address for exported datagrams.
Step 8	**output-features** **Example:** `Device(config-flow-exporter)# output-features`	(Optional) Enables sending export packets using quality of service (QoS) and encryption.
Step 9	**template data timeout** *seconds* **Example:**	(Optional) Configures resending of templates based on a timeout.

	Command or Action	Purpose
	`Device(config-flow-exporter)# template data timeout 120`	• The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours).
Step 10	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 650`	Specifies the UDP port on which the destination system is listening for exported datagrams. • The range for the *udp-port* argument is from 1 to 65536.
Step 11	**ttl** *seconds* **Example:** `Device(config-flow-exporter)# ttl 15`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. • The range for the *seconds* argument is from 1 to 255.
Step 12	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flow exporter configuration mode and returns to privileged EXEC mode.
Step 13	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 14	**show running-config flow exporter** *exporter-name* **Example:** `Device# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Configuration Examples for Flexible NetFlow v9 Export Format

Example: Configuring NetFlow v9 Export Format

The following example shows how to configure version 9 export for Flexible NetFlow.

This example starts in global configuration mode.

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
 export-protocol netflow-v9
 transport udp 90
 exit
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
```

```
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exporter EXPORTER-1
!
ip cef
!
interface GigabitEthernet 0/0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input
!
```

Additional Reference for Flexible NetFlow v9 Export Format

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Command List, All Releases*
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards

Standard	Title
None	—

MIBs

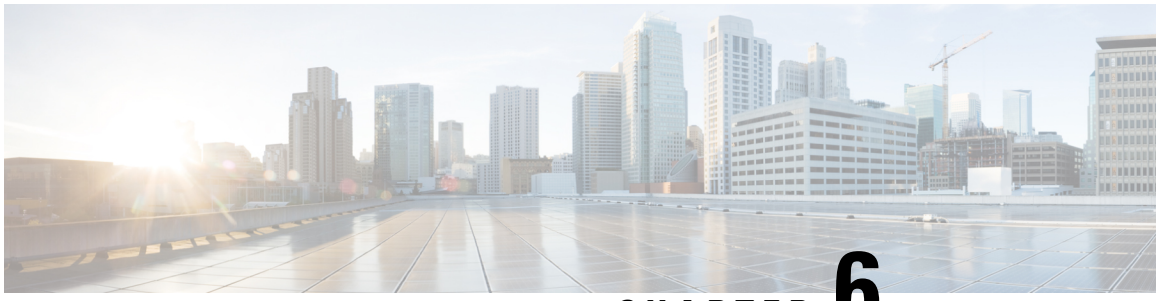
MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 3954	*Cisco Systems NetFlow Services Export Version 9*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 6

Flexible NetFlow Output Features on Data Export

This feature enables sending export packets using Quality of Service (QoS) and encryption.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Prerequisites for Flexible NetFlow Output Features on Data Export

- The networking device must be running a Cisco release that supports Flexible NetFlow.

IPv4 Traffic

- The networking device must be configured for IPv4 routing.
- One of the following must be enabled on your router and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding or distributed Cisco Express Forwarding.

IPv6 Traffic

- The networking device must be configured for IPv6 routing.
- One of the following must be enabled on your router and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding IPv6 or distributed Cisco Express Forwarding.

Information About Flexible NetFlow Output Features on Data Export

Flow Exporters

Flow exporters are created as separate components in a router's configuration. Exporters are assigned to flow monitors to export the data from the flow monitor cache to a remote system such as a NetFlow collector. Flow monitors can support more than one exporter. Each exporter can be customized to meet the requirements of the flow monitor or monitors in which it is used and the NetFlow collector systems to which it is exporting data.

Benefits of Flexible NetFlow Flow Exporters

Flexible NetFlow allows you to configure many different flow exporters, depending on your requirements. Some of the benefits of Flexible NetFlow flow exporters are as follows:

- Using flow exporters, you can create an exporter for every type of traffic that you want to analyze so that you can send each type of traffic to a different NetFlow collector. Original NetFlow sends the data in a cache for all of the analyzed traffic to a maximum of two export destinations.
- Flow exporters support up to ten exporters per flow monitor. Original NetFlow is limited to only two export destinations per cache.
- Flow exporters can use both TCP and UDP for export.
- Depending on your release, flow exporters can use class of service (CoS) in the packets that are sent to export destinations to help ensure that the packets are given the correct priority throughout the network. Original NetFlow exporters do not use CoS in the packets that are sent to export destinations.
- Depending on your release, flow exporter traffic can be encrypted.

How to Configure Flexible NetFlow Output Features on Data Export

Restrictions

Each flow exporter supports only one destination. If you want to export the data to multiple destinations, you must configure multiple flow exporters and assign them to the flow monitor. Flow exporters are added to flow monitors to enable data export from the flow monitor cache.

Note Only the keywords and arguments required for the Flexible NetFlow commands used in these tasks are explained in these tasks. For information about the other keywords and arguments available for these Flexible NetFlow commands, refer to the *Cisco IOS Flexible NetFlow Command Reference* .

Configuring the Flow Exporter

Perform this required task to configure the flow exporter.

Note Each flow exporter supports only one destination.

You can export to a destination using either an IPv4 or IPv6 address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **dscp** *dscp*
7. **source** *interface-type interface-number*
8. **output-features**
9. **template data timeout** *seconds*
10. **transport udp** *udp-port*
11. **ttl** *seconds*
12. **end**
13. **show flow exporter** *exporter-name*
14. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to the datacenter`	(Optional) Configures a description to the exporter that will appear in the configuration and the display of the **show flow exporter** command.
Step 5	**destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the IP address or hostname of the destination system for the exporter. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**dscp** *dscp* **Example:** `Device(config-flow-exporter)# dscp 63`	(Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter. • The range for the *dscp* argument is from 0 to 63. Default: 0.
Step 7	**source** *interface-type interface-number* **Example:** `Device(config-flow-exporter)# source ethernet 0/0`	(Optional) Specifies the local interface from which the exporter will use the IP address as the source IP address for exported datagrams.
Step 8	**output-features** **Example:** `Device(config-flow-exporter)# output-features`	(Optional) Enables sending export packets using quality of service (QoS) and encryption.
Step 9	**template data timeout** *seconds* **Example:**	(Optional) Configures resending of templates based on a timeout.

	Command or Action	Purpose
	`Device(config-flow-exporter)# template data timeout 120`	• The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours).
Step 10	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 650`	Specifies the UDP port on which the destination system is listening for exported datagrams. • The range for the *udp-port* argument is from 1 to 65536.
Step 11	**ttl** *seconds* **Example:** `Device(config-flow-exporter)# ttl 15`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. • The range for the *seconds* argument is from 1 to 255.
Step 12	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flow exporter configuration mode and returns to privileged EXEC mode.
Step 13	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 14	**show running-config flow exporter** *exporter-name* **Example:** `Device# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Displaying the Current Status of a Flow Exporter

To display the current status of a flow exporter, perform the following optional task.

SUMMARY STEPS

1. **enable**
2. **show flow exporter** [**export-ids {netflow-v5**| **netflow-v9**} | [**name**] *exporter-name* [**statistics** | **templates**]]

DETAILED STEPS

Step 1 **enable**

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 **show flow exporter** [**export-ids {netflow-v5**| **netflow-v9**} | [**name**] *exporter-name* [**statistics** | **templates**]]

The **show flow exporter**command shows the current status of the flow exporter that you specify.

Example:

```
Device# show flow exporter EXPORTER-1
Flow Exporter EXPORTER-1:
  Description:              Exports to the datacenter
  Export protocol:          NetFlow Version 9
  Transport Configuration:
    Destination IP address: 172.16.10.2
    Source IP address:      172.16.6.2
    Source Interface:       GigabitEthernet1/0/0
    Transport Protocol:     UDP
    Destination Port:       650
    Source Port:            55864
    DSCP:                   0x3F
    TTL:                    15
    Output Features:        Used
  Options Configuration:
    exporter-stats (timeout 120 seconds)
    interface-table (timeout 120 seconds)
    sampler-table (timeout 120 seconds)
```

Verifying the Flow Exporter Configuration

To verify the configuration commands that you entered, perform the following optional task.

SUMMARY STEPS

1. **enable**
2. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

Step 1 **enable**

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 **show running-config flow exporter** *exporter-name*

The **show running-config flow exporter** command shows the configuration commands of the flow exporter that you specify.

Example:

```
Device# show running-config flow exporter EXPORTER-1
Building configuration...
Current configuration:
!
flow exporter EXPORTER-1
 description Exports to the datacenter
 destination 172.16.10.2
 source GigabitEthernet1/0/0
 dscp 63
 ttl 15
 transport udp 650
 template data timeout 120
 option exporter-stats timeout 120
 option interface-table timeout 120
 option sampler-table timeout 120
!
end
```

Configuring and Enabling Flexible NetFlow with Data Export

You must create a flow monitor to configure the types of traffic for which you want to export the cache data. You must enable the flow monitor by applying it to at least one interface to start exporting data. To configure and enable Flexible NetFlow with data export, perform this required task.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. The record format can be one of the predefined record formats, or an advanced user may create his or her own record format using the **collect** and **match**commands in Flexible NetFlow flow record configuration mode.

Note You must remove a flow monitor from all of the interfaces to which you have applied it before you can modify the **record** format of the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6** *record* [**peer**] }]
5. **exporter** *exporter-name*
6. **exit**
7. **interface** *type number*
8. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
9. **end**
10. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.
Step 4	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6** *record* [**peer**] }] **Example:** `Device(config-flow-monitor)# record netflow ipv4 original-input`	Specifies the record for the flow monitor.
Step 5	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter EXPORTER-1`	Specifies the name of an exporter that you created previously.
Step 6	**exit** **Example:** `Device(config-flow-monitor)# exit`	Exits Flexible NetFlow flow monitor configuration mode and returns to global configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 8	{**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**} **Example:** `Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	Activates the flow monitor that you created previously by assigning it to the interface to analyze traffic.
Step 9	**end** **Example:**	Exits interface configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
	`Device(config-if)# end`	
Step 10	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]] **Example:** `Device# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor. This will verify data export is enabled for the flow monitor cache.

Configuration Examples for Flexible NetFlow Output Features on Data Export

Example: Configuring Sending Export Packets Using QoS

The following example shows how to enable QoS on Flexible Netflow export packets.

Note The Flexible NetFlow export packets are transmitted using QoS on Ethernet interface 0/1 (the interface on which the destination is reachable) to the destination host (IP address 10.0.1.2).

This sample starts in global configuration mode:

```
!
flow record FLOW-RECORD-1
 match ipv4 source address
 collect counter packets
!
flow exporter FLOW-EXPORTER-1
 destination 10.0.1.2
 output-features
 dscp 18
!
flow monitor FLOW-MONITOR-1
 record FLOW-RECORD-1
 exporter FLOW-EXPORTER-1
 cache entries 1024
!
ip cef
!
class-map match-any COS3
!
policy-map PH_LABS_FRL_64k_16k_16k_8k_8k
 class COS3
  bandwidth percent 2
  random-detect dscp-based
  random-detect exponential-weighting-constant 1
  random-detect dscp 18 200 300 10
!
interface Ethernet 0/0
 ip address 10.0.0.1 255.255.255.0
```

```
 ip flow monitor FLOW-MONITOR-1 input
!
interface Ethernet 0/1
 ip address 10.0.1.1 255.255.255.0
 service-policy output PH_LABS_FRL_64k_16k_16k_8k_8k
!
```

The following display output shows that the flow monitor is exporting data using output feature support that enables the exported data to use QoS:

```
Device# show flow monitor FLOW-MONITOR-1
Flow Exporter FLOW-EXPORTER-1:
  Description:              User defined
  Tranport Configuration:
    Destination IP address: 10.0.1.2
    Source IP address:      10.0.0.1
    Transport Protocol:     UDP
    Destination Port:       9995
    Source Port:            56750
    DSCP:                   0x12
    TTL:                    255
    Output Features:        Used
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

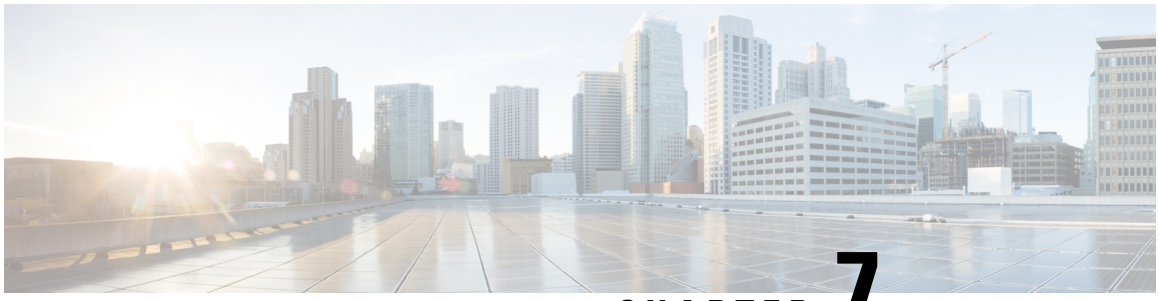
Feature Information for Flexible NetFlow—Output Features on Data Export

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

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

Table 21: Feature Information for Flexible NetFlow—Output Features on Data Export

Feature Name	Releases	Feature Information
Flexible NetFlow—Output Features on Data Export	12.4(20)T Cisco IOS XE Release 3.1S	Enables sending export packets using QoS and encryption. The following command was introduced: **output-features**.



CHAPTER 7

Flexible NetFlow NetFlow V5 Export Protocol

The Flexible Netflow NetFlow V5 Export Protocol feature enables sending export packets using the Version 5 export protocol.

Support for this feature was added for Cisco 7200 and 7300 Network Processing Engine (NPE) series routers in Cisco IOS Release 12.2(33)SRE.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Restrictions for Flexible NetFlow NetFlow V5 Export Protocol

- The NetFlow Version 5 export protocol that was first shipped in Cisco IOS Release 12.4(22)T is supported for flow monitors that use only the following Flexible NetFlow predefined records: netflow-original, original input, and original output.

Information about Flexible NetFlow NetFlow V5 Export Protocol

Flexible NetFlow V5 Export Protocol Overview

This feature enables sending export packets using the Version 5 export protocol.

How to Configure Flexible NetFlow NetFlow V5 Export Protocol

Configuring the Flow Exporter

Perform this required task to configure the flow exporter.

Note Each flow exporter supports only one destination.

You can export to a destination using either an IPv4 or IPv6 address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **dscp** *dscp*
7. **source** *interface-type interface-number*
8. **output-features**
9. **template data timeout** *seconds*
10. **transport udp** *udp-port*
11. **ttl** *seconds*
12. **end**
13. **show flow exporter** *exporter-name*
14. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to the datacenter`	(Optional) Configures a description to the exporter that will appear in the configuration and the display of the **show flow exporter** command.
Step 5	**destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the IP address or hostname of the destination system for the exporter. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**dscp** *dscp* **Example:** `Device(config-flow-exporter)# dscp 63`	(Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter. • The range for the *dscp* argument is from 0 to 63. Default: 0.
Step 7	**source** *interface-type interface-number* **Example:** `Device(config-flow-exporter)# source ethernet 0/0`	(Optional) Specifies the local interface from which the exporter will use the IP address as the source IP address for exported datagrams.
Step 8	**output-features** **Example:** `Device(config-flow-exporter)# output-features`	(Optional) Enables sending export packets using quality of service (QoS) and encryption.
Step 9	**template data timeout** *seconds* **Example:** `Device(config-flow-exporter)# template data timeout 120`	(Optional) Configures resending of templates based on a timeout. • The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours).
Step 10	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 650`	Specifies the UDP port on which the destination system is listening for exported datagrams. • The range for the *udp-port* argument is from 1 to 65536.

	Command or Action	Purpose
Step 11	**ttl** *seconds* **Example:** `Device(config-flow-exporter)# ttl 15`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. • The range for the *seconds* argument is from 1 to 255.
Step 12	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flow exporter configuration mode and returns to privileged EXEC mode.
Step 13	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 14	**show running-config flow exporter** *exporter-name* **Example:** `Device# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Configuration Examples for Flexible NetFlow NetFlow V5 Export Protocol

Example: Configuring Version 5 Export

The following example shows how to configure version 5 export for Flexible NetFlow.

This sample starts in global configuration mode:

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
 export-protocol netflow-v5
 transport udp 90
 exit
!
flow monitor FLOW-MONITOR-1
 record netflow ipv4 original-input
 exporter EXPORTER-1
!
ip cef
!
interface Ethernet 0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input
!
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

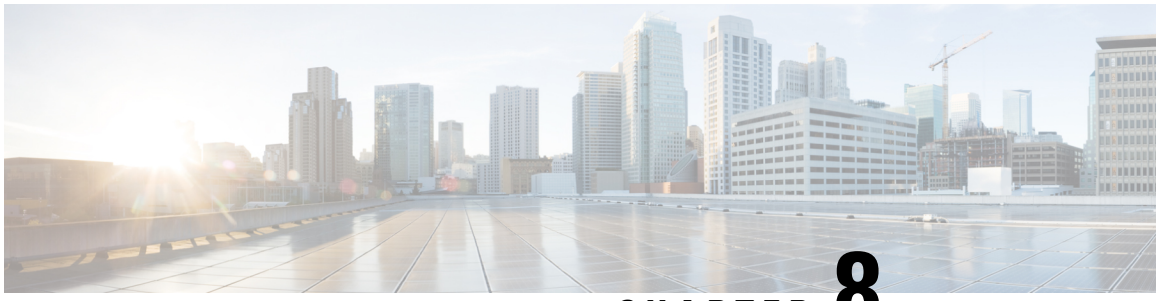
Feature Information for Flexible NetFlow NetFlow V5 Export Protocol

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

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

Table 22: Feature Information for Flexible NetFlow NetFlow V5 Export Protocol

Feature Name	Releases	Feature Information
Flexible NetFlow--NetFlow V5 Export Protocol	12.2(33)SRE 12.2(50)SY 12.4(22)T 15.0(1)SY 15.0(1)SY1 Cisco IOS XE Release 3.1S	Enables sending export packets using the Version 5 export protocol. Support for this feature was added for Cisco 7200 and 7300 Network Processing Engine (NPE) series routers in Cisco IOS Release 12.2(33)SRE. The following command was introduced: **export-protocol**.



CHAPTER 8

Using Flexible NetFlow Flow Sampling

This document contains information about and instructions for configuring sampling to reduce the CPU overhead of analyzing traffic with Flexible NetFlow.

NetFlow is a Cisco technology that provides statistics on packets flowing through a router. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides data to support network and security monitoring, network planning, traffic analysis, and IP accounting.

Flexible NetFlow improves on original NetFlow by adding the capability to customize the traffic analysis parameters for your specific requirements. Flexible NetFlow facilitates the creation of more complex configurations for traffic analysis and data export through the use of reusable configuration components.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Prerequisites for Using Flexible NetFlow Flow Sampling

- The networking device must be running a Cisco release that supports Flexible NetFlow.

Restrictions for Using Flexible NetFlow Flow Sampling

Information About Flexible NetFlow Flow Sampling

Flow Samplers

Flow samplers are created as separate components in a router's configuration. Flow samplers are used to reduce the load on the device that is running by limiting the number of packets that are selected for analysis.

Flow sampling exchanges monitoring accuracy for router performance. When you apply a sampler to a flow monitor, the overhead load on the router of running the flow monitor is reduced because the number of packets that the flow monitor must analyze is reduced. The reduction in the number of packets that are analyzed by the flow monitor causes a corresponding reduction in the accuracy of the information stored in the flow monitor's cache.

Samplers are combined with flow monitors when they are applied to an interface with the **ip flow monitor** command.

How to Configure Flexible NetFlow Flow Sampling

Flow sampling reduces the CPU overhead of analyzing traffic with Flexible NetFlow by reducing the number of packets that are analyzed.

Note Only the keywords and arguments required for the Flexible NetFlow commands used in these tasks are explained in these tasks. For information about the other keywords and arguments available for these Flexible NetFlow commands, refer to the *Cisco IOS Flexible NetFlow Command Reference* .

Configuring a Flow Monitor

Samplers are applied to an interface in conjunction with a flow monitor. You must create a flow monitor to configure the types of traffic that you want to analyze before you can enable sampling. Perform this required task to configure a flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. The record format can be one of the predefined record formats, or an advanced user may create his or her own record format using the **collect** and **match** commands in Flexible NetFlow flow record configuration mode.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.
Step 4	**description** *description* **Example:** `Device(config-flow-monitor)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `Device(config-flow-monitor)# record netflow ipv4 original-input`	Specifies the record for the flow monitor.
Step 6	**end** **Example:** `Device(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.

Configuring and Enabling Flow Sampling

Perform this required task to configure and enable a flow sampler.

Note When you specify the "NetFlow original," or the "NetFlow IPv4 original input," or the "NetFlow IPv6 original input" predefined record for the flow monitor to emulate original NetFlow, the flow monitor can be used only for analyzing input (ingress) traffic.

When you specify the "NetFlow IPv4 original output" or the "NetFlow IPv6 original output" predefined record for the flow monitor to emulate the Egress NetFlow Accounting feature, the flow monitor can be used only for analyzing output (egress) traffic.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **sampler** *sampler-name*
4. **description** *description*
5. **mode** {**random**} **1 out-of** *window-size*
6. **exit**
7. **interface** *type number*
8. {**ip** | **ipv6**} **flow monitor** *monitor-name* [[**sampler**] *sampler-name*] {**input** | **output**}
9. **end**
10. **show sampler sampler-name**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**sampler** *sampler-name* **Example:** `Device(config)# sampler SAMPLER-1`	Creates a sampler and enters sampler configuration mode. • This command also allows you to modify an existing sampler.
Step 4	**description** *description* **Example:** `Device(config-sampler)# description Sample at 50%`	(Optional) Creates a description for the flow sampler.
Step 5	**mode** {**random**} **1 out-of** *window-size* **Example:**	Specifies the sampler mode and the flow sampler window size.

	Command or Action	Purpose
	`Device(config-sampler)# mode random 1 out-of 2`	• The range for the *window-size* argument is from 2 to 32768.
Step 6	**exit** **Example:** `Device(config-sampler)# exit`	Exits sampler configuration mode and returns to global configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 8	{**ip** | **ipv6**} **flow monitor** *monitor-name* [[**sampler**] *sampler-name*] {**input** | **output**} **Example:** `Device(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-1 input`	Assigns the flow monitor and the flow sampler that you created to the interface to enable sampling.
Step 9	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 10	**show sampler sampler-name** **Example:** `Device# show sampler SAMPLER-1`	Displays the status and statistics of the flow sampler that you configured and enabled.

Displaying the Status and Statistics of the Flow Sampler Configuration

To display the status and statistics of the flow sampler that you configured and enabled, perform the following optional task.

SUMMARY STEPS

1. **enable**
2. **show sampler sampler-name**

DETAILED STEPS

Step 1 **enable**

The **enable** command enters privileged EXEC mode (enter the password if prompted).

Example:

```
Device> enable
Device#
```

Step 2 **show sampler sampler-name**

The **show sampler** command shows the current status of the sampler that you specify.

Example:

```
Device# show sampler SAMPLER-1
Sampler SAMPLER-1:
  ID:             2
  Description:    Sample at 50%
  Type:           random
  Rate:           1 out of 2
  Samples:        2482
  Requests:       4964
  Users (1):
    flow monitor FLOW-MONITOR-1 (ip,Et0/0,I  2482 out of 4964
```

Configuration Examples for Flexible NetFlow Flow Sampling

Example: Configuring and Enabling a Random Sampler for IPv4 Traffic

The following example shows how to configure and enable random sampling for IPv4 output traffic.

This example starts in global configuration mode.

```
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exit
!
sampler SAMPLER-1
 mode random 1 out-of 2
 exit
!
ip cef
!
interface GigabitEthernet 0/0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-1 output
!
```

The following example shows how to configure and enable random sampling for IPv4 input traffic.

This example starts in global configuration mode.

```
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!
flow monitor FLOW-MONITOR-1
 record v4_r1
 exit
!
sampler SAMPLER-1
 mode random 1 out-of 2
 exit
!
ip cef
!
interface GigabitEthernet 0/0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-1 input
!
```

Example: Adding a Sampler to a Flow Monitor When a Flow Monitor Is Already Enabled

The following example shows what happens when you try to add a sampler to a flow monitor that has already been enabled on an interface without a sampler:

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-2 input
% Flow Monitor: Flow Monitor 'FLOW-MONITOR-1' is already on in full mode and cannot be
enabled with a sampler.
```

The following example shows how to remove the flow monitor from the interface so that it can be enabled with the sampler:

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# no ip flow monitor FLOW-MONITOR-1 input
Device(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-2 input
```

Example: Removing a Sampler from a Flow Monitor

The following example shows what happens when you try to remove a sampler from a flow monitor on an interface by entering the **ip flow monitor** command again without the sampler keyword and argument:

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip flow monitor FLOW-MONITOR-1 input
% Flow Monitor: Flow Monitor 'FLOW-MONITOR-1' is already on in sampled mode and cannot be
enabled in full mode.
```

The following example shows how to remove the flow monitor that was enabled with a sampler from the interface so that it can be enabled without the sampler:

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# no ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-2 input
Device(config-if)# ip flow monitor FLOW-MONITOR-1 input
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

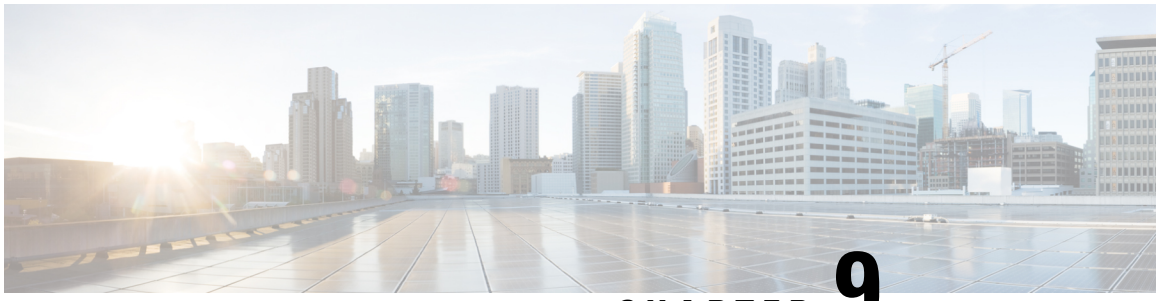
Feature Information for Flexible NetFlow Flow Sampling

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

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

Table 23: Feature Information for Flexible Netflow Flow Sampling

Feature Name	Releases	Feature Information
Flexible Netflow - Random Sampling	12.2(50)SY 12.4(20)T Cisco IOS XE Release 3.1S Cisco IOS XE Release 3.2SE	Flow samplers are created as separate components in a router's configuration. Flow samplers are used to reduce the load on the device that is running Flexible NetFlow by limiting the number of packets that are selected for analysis. Samplers use either random or deterministic sampling techniques (modes). The following commands were introduced or modified: **clear sampler**, **debug sampler** , **mode** , **record** , **sampler** , **show sampler** .



CHAPTER 9

Configuring IPv4 Multicast Statistics Support for Flexible NetFlow

This document contains information about and instructions for configuring the Cisco IOS Flexible NetFlow - IPv4 Multicast Statistics Support feature. Prior to the introduction of the Flexible NetFlow - IPv4 Multicast Statistics Support feature, Flexible NetFlow could analyze IPv4 multicast traffic, but could not report the number of replicated bytes or the number of replicated packets in multicast flows. The Flexible NetFlow - IPv4 Multicast Statistics Support feature adds the capability of reporting the number of replicated bytes and the number of replicated packets in multicast flows to Flexible NetFlow.

NetFlow is a Cisco IOS technology that provides statistics on packets flowing through a networking device. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides provides data to support network and security monitoring, network planning, traffic analysis, and IP accounting.

Flexible NetFlow improves on original NetFlow by adding the capability to customize the traffic analysis parameters for your specific requirements. Flexible NetFlow facilitates the creation of more complex configurations for traffic analysis and data export through the use of reusable configuration components.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Prerequisites for Configuring IPv4 Multicast Statistics Support

- The networking device is running a Cisco IOS release that supports the Flexible NetFlow--IPv4 Multicast Statistics Support feature.
- The networking device is configured for IPv4 unicast routing and IPv4 multicast routing.
- One of the following is enabled on your networking device and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding, distributed Cisco Express Forwarding.

Restrictions for Configuring IPv4 Multicast Statistics Support

IPv4 Traffic

- When the replication-factor field is used in a flow record, it will only have a nonzero value in the cache for ingress multicast traffic that is forwarded by the router. If the flow record is used with a flow monitor in output (egress) mode and to monitor unicast traffic, the cache data for the replication factor field is set to 0.

IPv6 Traffic

- Traffic monitoring for multicast statistics is not supported.

Note The**match routing multicast replication-factor** command is not supported on ASR and ISR platforms.

Information About IPv4 Multicast Statistics Support

Replicated Bytes and Packets Reporting

The Flexible NetFlow--IPv4 Multicast Statistics Support feature adds the capability of reporting the number of replicated bytes and the number of replicated packets in multicast flows to Flexible NetFlow. You can capture the packet-replication factor for a specific flow and for each outgoing stream.

You can use the The Flexible NetFlow--IPv4 Multicast Statistics Support feature to identify and count multicast packets on the ingress side or the egress side (or both sides) of a networking device. Multicast ingress accounting provides information about the source and how many times the traffic was replicated. Multicast egress accounting monitors the destination of the traffic flow.

How to Configure IPv4 Multicast Statistics Support

Configuring IPv4 Multicast Statistics Support

This task explains the steps that are used to configure multicast statistics support for IPv4 traffic.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *flow-record-name*
4. **description** *description*
5. **match routing is-multicast**
6. Add key fields for the record as required using other **match** commands.
7. **collect counter** {**bytes replicated** [**long**] | **packets replicated** [**long**]}
8. **collect routing multicast replication-factor**
9. Add nonkey fields for the record as required using other **collect** commands.
10. **exit**
11. **flow monitor** *monitor-name*
12. **description** *description*
13. **record** *record-name*
14. **exit**
15. **interface** *type number*
16. **ip flow monitor** *monitor-name* [**multicast** | **unicast**] {**input** | **output**}
17. Repeat Steps 15 and 16 to activate a flow monitor on any other interfaces in the networking device over which you want to monitor traffic.
18. **end**
19. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *flow-record-name* **Example:**	Creates a flow record and enters Flexible NetFlow flow record configuration mode.

	Command or Action	Purpose
	`Device(config)# flow record FLOW-RECORD-2`	• This command also allows you to modify an existing flow record.
Step 4	**description** *description* **Example:** `Device(config-flow-record)# description Used for IPv4 multicast traffic analysis`	(Optional) Creates a description for the flow record.
Step 5	**match routing is-multicast** **Example:** `Device(config-flow-record)# match routing is-multicast`	Configures IPv4 multicast destination addresses (indicating that the IPv4 traffic is multicast traffic) as a key field for the flow record.
Step 6	Add key fields for the record as required using other **match** commands.	For information about the other **match** commands that are available to configure key fields, refer to the *Cisco IOS Flexible NetFlow Command Reference*.
Step 7	**collect counter** {**bytes replicated** [**long**] | **packets replicated** [**long**]} **Example:** `Device(config-flow-record)# collect counter packets replicated`	Configures the number of bytes or packets multiplied by the multicast replication factor (number of interfaces the multicast traffic is forwarded over) as a nonkey field. • Default: Uses a 32-bit counter. The **long** keyword configures a 64-bit counter.
Step 8	**collect routing multicast replication-factor** **Example:** `Device(config-flow-record)# collect routing multicast replication-factor`	Configures the multicast replication factor (number of interfaces over which multicast traffic is forwarded) as a nonkey field.
Step 9	Add nonkey fields for the record as required using other **collect** commands.	For information about the other **collect** commands that are available to configure nonkey fields, refer to the *Cisco IOS Flexible NetFlow Command Reference*.
Step 10	**exit** **Example:** `Device(config-flow-record)# exit`	Exits Flexible NetFlow flow record configuration mode and returns to global configuration mode.
Step 11	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-2`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.
Step 12	**description** *description* **Example:**	(Optional) Creates a description for the flow monitor.

	Command or Action	Purpose
	`Device(config-flow-monitor)# description Used for IPv4 multicast traffic analysis`	
Step 13	**record** *record-name* **Example:** `Device(config-flow-monitor)# record FLOW-RECORD-2`	Specifies the record for the flow monitor.
Step 14	**exit** **Example:** `Device(config-flow-record)# exit`	Exits Flexible NetFlow flow monitor configuration mode and returns to global configuration mode.
Step 15	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 16	**ip flow monitor** *monitor-name* [**multicast** | **unicast**] {**input** | **output**} **Example:** `Device(config-if)# ip flow monitor FLOW-MONITOR-2 input`	Activates the flow monitor that was created previously by assigning it to the interface to analyze traffic. • To monitor only multicast traffic, use the **multicast** keyword. • Default: Unicast traffic and multicast traffic are monitored.
Step 17	Repeat Steps 15 and 16 to activate a flow monitor on any other interfaces in the networking device over which you want to monitor traffic.	--
Step 18	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 19	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]][**statistics**]] **Example:** `Device# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.

Examples

The following output from the **show flow monitor** command shows four multicast flows and three unicast flows:

```
Device# show flow monitor FLOW-MONITOR-2 cache

Cache type:                               Normal
  Cache size:                               4096
  Current entries:                             8
  High Watermark:                              8
  Flows added:                              4074
  Flows aged:                               4066
    - Active timeout   (  1800 secs)          46
    - Inactive timeout (    15 secs)        4020
    - Event aged                                0
    - Watermark aged                            0
    - Emergency aged                            0
IP IS MULTICAST  IPV4 DST ADDR      pkts rep
===============  ===============  ==========
Yes              224.192.16.1          16642
Yes              224.192.65.1          16621
No               10.1.4.2                  0
No               10.1.2.2                  0
No               10.1.3.2                  0
Yes              224.0.0.13                0
No               255.255.255.255           0
Yes              224.0.0.1                 0
```

Configuration Examples for IPv4 Multicast Statistics Support

Example: Configuring IPv4 Multicast Statistics Support

This example shows how to configure the following:

- IPv4 multicast destination addresses (indicating that the IPv4 traffic is multicast traffic) as a key field.
- The destination IPv4 address as a key field.
- The replicated packet count as a nonkey field.
- The replication factor as a nonkey field.
- The flow monitor in order to monitor only multicast traffic.

This sample starts in global configuration mode:

```
!
flow record FLOW-RECORD-2
 match routing is-multicast
 match ipv4 destination address
 collect counter packets replicated
 collect routing multicast replication-factor
 exit
!
flow monitor FLOW-MONITOR-2
 record FLOW-RECORD-2
 exit
!
interface GigabitEthernet 0/0/0
 no shut
 ip address 10.1.1.2 255.255.255.0
```

```
 ip flow monitor FLOW-MONITOR-2 multicast input
!
end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

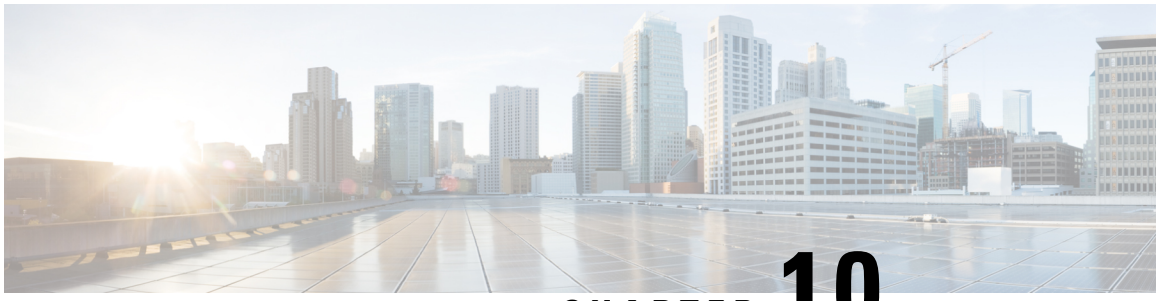
Feature Information for IPv4 Multicast Statistics Support

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

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

Table 24: Feature Information for Flexible NetFlow --IPv4 Multicast Statistics Support

Feature Name	Releases	Feature Information
Flexible NetFlow--IPv4 Multicast Statistics Support	12.2(33)SRE 12.2(50)SY 12.4(22)T	The Flexible NetFlow--IPv4 Multicast Statistics Support feature adds the capability of reporting the number of replicated bytes and the number of replicated packets in multicast flows to Flexible NetFlow. Support for this feature was added for Cisco 7200 and 7300 Network Processing Engine (NPE) series routers in Cisco IOS Release 12.2(33)SRE. The following commands were introduced or modified: **collect counter**, **collect routing is-multicast**, **collect routing multicast replication-factor**, **match routing is-multicast**, **match routing multicast replication-factor**, **ip flow monitor**, **ipv6 flow monitor**.



CHAPTER 10

Flexible NetFlow - Top N Talkers Support

This document contains information about and instructions for using the Flexible NetFlow - Top N Talkers Support feature. The Flexible NetFlow - Top N Talkers Support feature helps you analyze the large amount of data that Flexible NetFlow captures from the traffic in your network by providing the ability to filter, aggregate, and sort the data in the Flexible NetFlow cache as you display it. When you are sorting and displaying the data in the cache, you can limit the display output to a specific number of entries with the highest values (Top N Talkers) for traffic volume, packet counters, and so on. The Flexible NetFlow - Top N Talkers Support feature facilitates real-time traffic analysis by requiring only the use of **show** commands, which can be entered in many different variations using the available keywords and arguments to meet your traffic data analysis requirements.

NetFlow is a Cisco technology that provides statistics on packets flowing through the router. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides data to support network and security monitoring, network planning, traffic analysis, and IP accounting.

Flexible NetFlow improves on original NetFlow by adding the capability to customize the traffic analysis parameters for your specific requirements. Flexible NetFlow facilitates the creation of more complex configurations for traffic analysis and data export through the use of reusable configuration components.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Prerequisites for Flexible NetFlow - Top N Talkers Support

- The networking device is running a Cisco release that supports the Flexible NetFlow - Top N Talkers Support feature.

No configuration tasks are associated with the Flexible NetFlow - Top N Talkers Support feature. Therefore, in order for you to use the Flexible NetFlow - Top N Talkers Support feature, traffic analysis with Flexible NetFlow must already be configured on the networking device.

Information About Flexible NetFlow - Top N Talkers Support

Flexible NetFlow Data Flow Filtering

The flow filtering function of the Flexible NetFlow - Top N Talkers Support feature filters the flow data in a flow monitor cache based on the criteria that you specify, and displays the data.

The flow filtering function of the Flexible NetFlow - Top N Talkers Support feature is provided by the **show flow monitor cache filter** command. For more information on the **show flow monitor cache filter** command, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

Flexible NetFlow Data Flow Aggregation

Flow aggregation using the **show flow monitor cache aggregate** command allows you to dynamically view the flow information in a cache using a different flow record than the cache was originally created from. Only the fields in the cache will be available for the aggregated flows.

The flow aggregation function of the Flexible NetFlow - Top N Talkers Support feature is provided by the **show flow monitor cache aggregate** command. For more information on the **show flow monitor cache aggregate** command, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

Flow Sorting and Top N Talkers

The flow sorting function of the Flexible NetFlow - Top N Talkers Support feature sorts flow data from the Flexible NetFlow cache based on the criteria that you specify and displays the data. You can also use the flow sorting function of the Flexible NetFlow - Top N Talkers Support feature to limit the display output to a specific number of entries (top *n* talkers, where *n* is the number or talkers to display) by using the **top** keyword of the **show flow monitor cache sort** command.

The flow sorting and Top N Talkers function of the Flexible NetFlow - Top N Talkers Support feature is provided by the **show flow monitor cache sort** command. For more information on the **show flow monitor cache sort** command, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

Combined Use of Flow Filtering and Flow Aggregation and Flow Sorting with Top N Talkers

Although each of the **show** commands that make up the Flexible NetFlow - Top N Talkers Support feature can be used individually for traffic analysis, they provide much greater analytical capabilities when they are used together. When you use any combination of the three **show** commands, you enter only the common prefix of **show flow monitor** *monitor-name* **cache**followed by **filter**, **aggregation**, or **sort**, and the arguments and keywords available for **filter**, **aggregation**, and **sort**, as required. For example,

```
show flow monitor
monitor-name
cache filter

options
 aggregation
options
 sort
options
```

where *options* is any permissible combination of arguments and keywords. See the "Configuration Examples for Flexible NetFlow - Top N Talkers Support " section for more information.

Memory and Performance Impact of Top N Talkers

The Flexible NetFlow - Top N Talkers Support feature can use a large number of CPU cycles and possibly also system memory for a short time. However, because the Flexible NetFlow - Top N Talkers Support feature uses only **show** commands, the CPU usage should be run at a low priority because no real-time data processing is involved. The memory usage can be mitigated by using a larger granularity of aggregation or no aggregation at all.

How to Analyze Network Traffic Using Flexible NetFlow Top N Talkers

Filtering Flow Data from the Flexible NetFlow Cache

This task shows you how to use the **show flow monitor cache filter** command with a regular expression to filter the flow monitor cache data and display the results. For more information on regular expressions and the **show flow monitor cache filter** command, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

Perform this task to filter the flow monitor cache data using a regular expression and display the results.

SUMMARY STEPS

1. **enable**
2. **show flow monitor** [**name**] *monitor-name* **cache filter** *options* [**regexp** *regexp*] [*...options* [**regexp** *regexp*]] [**format** {**csv** | **record** | **table**}]

DETAILED STEPS

Step 1 **enable**

Enters privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 **show flow monitor** [**name**] *monitor-name* **cache filter** *options* [**regexp** *regexp*] [*...options* [**regexp** *regexp*]] [**format** {**csv** | **record** | **table**}]

Filters the flow monitor cache data on the IPv4 type of service (ToS) value.

Example:

```
Device# show flow monitor FLOW-MONITOR-3 cache filter ipv4 tos regexp 0x(C0|50)

Cache type:                               Normal
  Cache size:                               4096
  Current entries:                            19
  High Watermark:                             38
  Flows added:                              3516
  Flows aged:                               3497
    - Active timeout   (  1800 secs)          52
    - Inactive timeout (    15 secs)        3445
    - Event aged                               0
    - Watermark aged                           0
    - Emergency aged                           0
IPV4 SOURCE ADDRESS:       10.1.1.1
IPV4 DESTINATION ADDRESS:  255.255.255.255
TRNS SOURCE PORT:          520
TRNS DESTINATION PORT:     520
INTERFACE INPUT:           Et0/0
FLOW SAMPLER ID:           0
IP TOS:                    0xC0
IP PROTOCOL:               17
ip source as:              0
ip destination as:         0
ipv4 next hop address:     0.0.0.0
ipv4 source mask:          /24
ipv4 destination mask:     /0
tcp flags:                 0x00
interface output:          Null
counter bytes:             52
counter packets:           1
timestamp first:           18:59:46.199
timestamp last:            18:59:46.199
Matched 1 flow
```

Aggregating Flow Data from the Flexible NetFlow Cache

This task shows you how to use the **show flow monitor cache aggregate** command to aggregate the flow monitor cache data with a different record than the cache was created with and display the results. For more information on the **show flow monitor cache aggregate** command, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

Perform this task to aggregate the flow monitor cache data and display the results.

SUMMARY STEPS

1. **enable**
2. **show flow monitor** [**name**] *monitor-name* **cache aggregate** {*options* [*...options*] [**collect** *options* [*...options*]] | **record** *record-name*} [**format** {**csv** | **record** | **table**}]

DETAILED STEPS

Step 1 **enable**

Enters privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 **show flow monitor** [**name**] *monitor-name* **cache aggregate** {*options* [*...options*] [**collect** *options* [*...options*]] | **record** *record-name*} [**format** {**csv** | **record** | **table**}]

Aggregates the flow monitor cache data on the IPv4 destination address and displays the cache data for the IPv4 protocol type and input interface nonkey fields:

Example:

```
Device# show flow monitor FLOW-MONITOR-3 cache aggregate ipv4 destination address collect ipv4 protocol
 interface input

Processed 17 flows
Aggregated to 7 flows
IPV4 DST ADDR    intf input                flows       bytes        pkts  ip prot
===============  ====================  ==========  ==========  ==========  =======
224.192.16.4     Et0/0                          3       42200        2110        1
224.192.16.1     Et0/0                          3       17160         858        1
224.192.18.1     Et0/0                          4       18180         909        1
224.192.45.12    Et0/0                          4       14440         722        1
255.255.255.255  Et0/0                          1          52           1       17
224.0.0.13       Et0/0                          1          54           1      103
224.0.0.1        Et0/0                          1          28           1        2
```

Sorting Flow Data from the Flexible NetFlow Cache

This task shows you how to use the **show flow monitor cache sort** command to sort the flow monitor cache data, and display the results. For more information on the **show flow monitor cache sort** command, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

Perform this task to sort the flow monitor cache data and display the results.

SUMMARY STEPS

1. **enable**
2. **show flow monitor** [**name**] *monitor-name* **cache sort** *options* [**top** [*number*]] [**format** {**csv** | **record** | **table**}]

DETAILED STEPS

Step 1 enable

Enters privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 show flow monitor [name] monitor-name cache sort options [top [number]] [format {csv | record | table}]

Displays the cache data sorted on the number of packets from highest to lowest.

Note When the **top** keyword is not used, the default number of sorted flows shown is 20.

Example:

```
Device# show flow monitor FLOW-MONITOR-1 cache sort highest counter packets

Processed 26 flows
Aggregated to 26 flows
Showing the top 20 flows
IPV4 SOURCE ADDRESS:       10.1.1.3
IPV4 DESTINATION ADDRESS:  172.16.10.11
TRNS SOURCE PORT:          443
TRNS DESTINATION PORT:     443
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               6
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             22760
counter packets:           1569
timestamp first:           19:42:32.924
timestamp last:            19:57:28.656
IPV4 SOURCE ADDRESS:       10.10.11.2
IPV4 DESTINATION ADDRESS:  172.16.10.6
```

```
TRNS SOURCE PORT:          65
TRNS DESTINATION PORT:     65
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               6
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             22720
counter packets:           568
timestamp first:           19:42:34.264
timestamp last:            19:57:28.428
.
.
.
IPV4 SOURCE ADDRESS:       192.168.67.6
IPV4 DESTINATION ADDRESS:  172.16.10.200
TRNS SOURCE PORT:          0
TRNS DESTINATION PORT:     3073
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               1
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             15848
counter packets:           344
timestamp first:           19:42:36.852
timestamp last:            19:57:27.836
IPV4 SOURCE ADDRESS:       10.234.53.1
IPV4 DESTINATION ADDRESS:  172.16.10.2
TRNS SOURCE PORT:          0
TRNS DESTINATION PORT:     2048
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               1
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             15848
counter packets:           213
timestamp first:           19:42:36.904
timestamp last:            19:57:27.888
```

Displaying the Top N Talkers with Sorted Flow Data

This task shows you how to use the **show flow monitor cache sort** command to sort the flow monitor cache data, and to limit the display results to a specific number of high volume flows. For more information on the **show flow monitor cache sort** command, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

Perform this task to sort the flow monitor cache data and limit the display output using to a specific number of high volume flows.

SUMMARY STEPS

1. **enable**
2. **show flow monitor** [**name**] *monitor-name* **cache sort** *options* [**top** [*number*]] [**format** {**csv** | **record** | **table**}]

DETAILED STEPS

Step 1 enable

Enters privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 show flow monitor [name] monitor-name cache sort options [top [number]] [format {csv | record | table}]

Displays the cache data sorted on the number of packets from highest to lowest and limits the output to the three highest volume flows.

Example:

```
Device# show flow monitor FLOW-MONITOR-1 cache sort highest counter packets top 3

Processed 25 flows
Aggregated to 25 flows
Showing the top 3 flows
IPV4 SOURCE ADDRESS:       10.1.1.3
IPV4 DESTINATION ADDRESS:  172.16.10.11
TRNS SOURCE PORT:          443
TRNS DESTINATION PORT:     443
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               6
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             32360
counter packets:           1897
timestamp first:           19:42:32.924
timestamp last:            20:03:47.100
IPV4 SOURCE ADDRESS:       10.10.11.2
IPV4 DESTINATION ADDRESS:  172.16.10.6
```

```
TRNS SOURCE PORT:          65
TRNS DESTINATION PORT:     65
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               6
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             32360
counter packets:           809
timestamp first:           19:42:34.264
timestamp last:            20:03:48.460
IPV4 SOURCE ADDRESS:       172.16.1.84
IPV4 DESTINATION ADDRESS:  172.16.10.19
TRNS SOURCE PORT:          80
TRNS DESTINATION PORT:     80
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               6
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /24
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             32320
counter packets:           345
timestamp first:           19:42:34.512
timestamp last:            20:03:47.140
```

Configuration Examples for Flexible NetFlow Top N Talkers

Example: Displaying the Top Talkers with Filtered and Aggregated and Sorted Flow Data

The following example combines filtering, aggregation, collecting additional field data, sorting the flow monitor cache data, and limiting the display output to a specific number of high volume flows (top talkers).

```
Device# show flow monitor FLOW-MONITOR-1 cache filter ipv4 protocol regexp (1|6) aggregate
 ipv4 destination address collect ipv4 protocol sort counter bytes top 4

Processed 26 flows
Matched 26 flows
Aggregated to 13 flows
Showing the top 4 flows
IPV4 DST ADDR         flows       bytes        pkts
===============  ==========  ==========  ==========
```

```
172.16.10.2              12     1358370        6708
172.16.10.19              2       44640        1116
172.16.10.20              2       44640        1116
172.16.10.4               1       22360         559
```

The following example combines filtering using a regular expression, aggregation using a predefined record, sorting the flow monitor cache data, limiting the display output to a specific number of high volume flows (top talkers), and displaying the output in record format.

```
Device# show flow monitor FLOW-MONITOR-1 cache filter ipv4 source address regexp 10.*
aggregate record netflow ipv4 protocol-port sort transport destination-port top 5 format
record

Processed 26 flows
Matched 15 flows
Aggregated to 10 flows
Showing the top 5 flows
TRNS SOURCE PORT:       0
TRNS DESTINATION PORT:  0
FLOW DIRECTION:         Input
IP PROTOCOL:            1
counter flows:          1
counter bytes:          387800
counter packets:        700
timestamp first:        17:12:30.712
timestamp last:         17:30:52.936
TRNS SOURCE PORT:       20
TRNS DESTINATION PORT:  20
FLOW DIRECTION:         Input
IP PROTOCOL:            6
counter flows:          2
counter bytes:          56000
counter packets:        1400
timestamp first:        17:12:29.532
timestamp last:         17:30:53.148
TRNS SOURCE PORT:       21
TRNS DESTINATION PORT:  21
FLOW DIRECTION:         Input
IP PROTOCOL:            6
counter flows:          2
counter bytes:          56000
counter packets:        1400
timestamp first:        17:12:29.572
timestamp last:         17:30:53.196
TRNS SOURCE PORT:       22
TRNS DESTINATION PORT:  22
FLOW DIRECTION:         Input
IP PROTOCOL:            6
counter flows:          1
counter bytes:          28000
counter packets:        700
timestamp first:        17:12:29.912
timestamp last:         17:30:52.168
TRNS SOURCE PORT:       25
TRNS DESTINATION PORT:  25
FLOW DIRECTION:         Input
IP PROTOCOL:            6
counter flows:          2
counter bytes:          56000
counter packets:        1400
timestamp first:        17:12:29.692
timestamp last:         17:30:51.968
```

Example: Filtering Using Multiple Filtering Criteria

The following example filters the cache data on the IPv4 destination address and the destination port:

```
Device# show flow monitor FLOW-MONITOR-1 cache filter ipv4 destination address regexp
172.16.10* transport destination-port 21

Cache type:                               Normal
  Cache size:                               4096
  Current entries:                            26
  High Watermark:                             26
  Flows added:                               241
  Flows aged:                                215
    - Active timeout   (  1800 secs)          50
    - Inactive timeout (    15 secs)         165
    - Event aged                              0
    - Watermark aged                          0
    - Emergency aged                          0
IPV4 SOURCE ADDRESS:       10.10.10.2
IPV4 DESTINATION ADDRESS:  172.16.10.2
TRNS SOURCE PORT:          21
TRNS DESTINATION PORT:     21
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               6
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             17200
counter packets:           430
timestamp first:           17:03:58.071
timestamp last:            17:15:14.615
IPV4 SOURCE ADDRESS:       172.30.231.193
IPV4 DESTINATION ADDRESS:  172.16.10.2
TRNS SOURCE PORT:          21
TRNS DESTINATION PORT:     21
INTERFACE INPUT:           Et0/0.1
FLOW SAMPLER ID:           0
IP TOS:                    0x00
IP PROTOCOL:               6
ip source as:              0
ip destination as:         0
ipv4 next hop address:     172.16.7.2
ipv4 source mask:          /0
ipv4 destination mask:     /24
tcp flags:                 0x00
interface output:          Et1/0.1
counter bytes:             17160
counter packets:           429
timestamp first:           17:03:59.963
timestamp last:            17:15:14.887
Matched 2 flows
```

Example: Aggregation Using Multiple Aggregation Criteria

The following example aggregates the flow monitor cache data on the destination and source IPv4 addresses:

```
Device# show flow monitor FLOW-MONITOR-1 cache aggregate ipv4 destination address ipv4
source address

Processed 26 flows
Aggregated to 17 flows
IPV4 SRC ADDR    IPV4 DST ADDR         flows       bytes        pkts
===============  ===============  ==========  ==========  ==========
10.251.10.1      172.16.10.2               2     1400828        1364
192.168.67.6     172.16.10.200             1       19096         682
10.234.53.1      172.16.10.2               3       73656        2046
172.30.231.193   172.16.10.2               3       73616        2045
10.10.10.2       172.16.10.2               2       54560        1364
192.168.87.200   172.16.10.2               2       54560        1364
10.10.10.4       172.16.10.4               1       27280         682
10.10.11.1       172.16.10.5               1       27280         682
10.10.11.2       172.16.10.6               1       27280         682
10.10.11.3       172.16.10.7               1       27280         682
10.10.11.4       172.16.10.8               1       27280         682
10.1.1.1         172.16.10.9               1       27280         682
10.1.1.2         172.16.10.10              1       27280         682
10.1.1.3         172.16.10.11              1       27280         682
172.16.1.84      172.16.10.19              2       54520        1363
172.16.1.85      172.16.10.20              2       54520        1363
172.16.6.1       224.0.0.9                 1          52           1
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

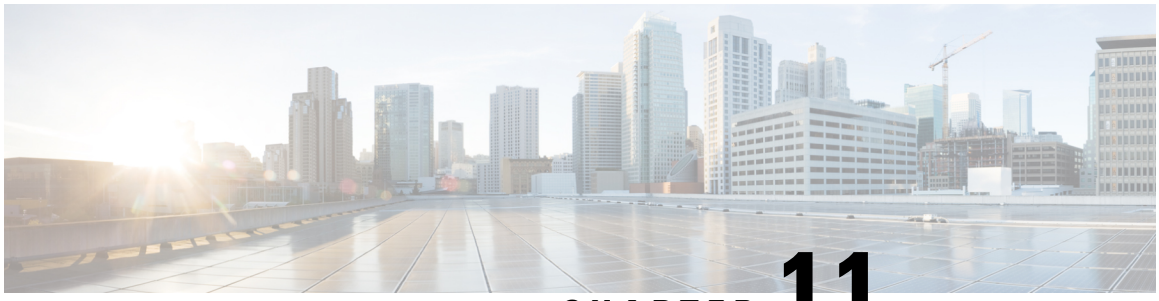
Feature Information for Flexible NetFlow - Top N Talkers

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

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

Table 25: Feature Information for Flexible NetFlow - Top N Talkers

Feature Name	Releases	Feature Information
Flexible NetFlow - Top N Talkers Support		This feature helps you analyze the large amount of data Flexible NetFlow captures from the traffic in your network by providing the ability to filter, aggregate, and sort the data in the Flexible NetFlow cache as you display it. Support for this feature was added for Cisco 7200 and 7300 Network Processing Engine (NPE) series routers in Cisco IOS Release 12.2(33)SRE. The following commands were introduced or modified: **show flow monitor cache aggregate**, **show flow monitor cache filter**, **show flow monitor cache**.



CHAPTER 11

Flexible NetFlow - Layer 2 Fields

The Flexible NetFlow - Layer 2 Fields feature enables collecting statistics for Layer 2 fields such as MAC addresses and virtual LAN (VLAN) IDs from traffic.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible NetFlow Layer 2 Fields

Flexible NetFlow - Layer 2 Fields Overview

The Flexible NetFlow - Layer 2 Fields feature enables collecting statistics for Layer 2 fields such as MAC addresses and virtual LAN (VLAN) IDs from traffic.

How to Configure Flexible NetFlow Layer 2 Fields

Creating a Customized Flow Record

Perform this task to configure a customized flow record.

Customized flow records are used to analyze traffic data for a specific purpose. A customized flow record must have at least one **match** criterion for use as the key field and typically has at least one **collect** criterion for use as a nonkey field.

There are hundreds of possible permutations of customized flow records. This task shows the steps that are used to create one of the possible permutations. Modify the steps in this task as appropriate to create a customized flow record for your requirements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **description** *description*
5. **match** {**ip** | **ipv6**} {**destination** | **source**} **address**
6. Repeat Step 5 as required to configure additional key fields for the record.
7. **collect interface** {**input** | **output**}
8. Repeat the above step as required to configure additional nonkey fields for the record.
9. **end**
10. **show flow record** *record-name*
11. **show running-config flow record** *record-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-RECORD-1`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. • This command also allows you to modify an existing flow record.

	Command or Action	Purpose
Step 4	**description** *description* **Example:** `Device(config-flow-record)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow record.
Step 5	**match** {**ip** | **ipv6**} {**destination** | **source**} **address** **Example:** `Device(config-flow-record)# match ipv4 destination address`	Configures a key field for the flow record. **Note** This example configures the IPv4 destination address as a key field for the record. For information about the other key fields available for the **match ipv4** command, and the other **match** commands that are available to configure key fields.
Step 6	Repeat Step 5 as required to configure additional key fields for the record.	—
Step 7	**collect interface** {**input** | **output**} **Example:**	Configures the input interface as a nonkey field for the record. **Note** This example configures the input interface as a nonkey field for the record.
Step 8	Repeat the above step as required to configure additional nonkey fields for the record.	—
Step 9	**end** **Example:** `Device(config-flow-record)# end`	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 10	**show flow record** *record-name* **Example:** `Device# show flow record FLOW_RECORD-1`	(Optional) Displays the current status of the specified flow record.
Step 11	**show running-config flow record** *record-name* **Example:** `Device# show running-config flow record FLOW_RECORD-1`	(Optional) Displays the configuration of the specified flow record.

Creating a Customized Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be one of the predefined formats or a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record instead of using one of the Flexible NetFlow predefined records, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}}
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **statistics packet protocol**
9. **statistics packet size**
10. **exporter** *exporter-name*
11. **end**
12. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]]
13. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:**	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode.

	Command or Action	Purpose
	`(config)# flow monitor FLOW-MONITOR-1`	• This command also allows you to modify an existing flow monitor.
Step 4	**description** *description* **Example:** `(config-flow-monitor)# description Used for basic ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}} **Example:**	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. The values for the keywords associated with the **timeout** keyword have no effect when the cache type is set to **immediate**.
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	**statistics packet protocol** **Example:** `(config-flow-monitor)# statistics packet protocol`	(Optional) Enables the collection of protocol distribution statistics for Flexible NetFlow monitors.
Step 9	**statistics packet size** **Example:** `(config-flow-monitor)# statistics packet size`	(Optional) Enables the collection of size distribution statistics for Flexible NetFlow monitors.
Step 10	**exporter** *exporter-name* **Example:** `(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.
Step 11	**end** **Example:** `(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] **Example:**	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.

	Command or Action	Purpose
	`# show flow monitor FLOW-MONITOR-2 cache`	
Step 13	**show running-config flow monitor** *monitor-name* **Example:** `# show running-config flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Applying a Flow Monitor to an Interface

Before it can be activated, a flow monitor must be applied to at least one interface. Perform this required task to activate a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
5. Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.
6. **end**
7. **show flow interface** *type number*
8. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 4	{**ip**	**ipv6**} **flow monitor** *monitor-name* {**input**	**output**} **Example:**	Activates a flow monitor that was created previously by assigning it to the interface to analyze traffic.

	Command or Action	Purpose
	`Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	
Step 5	Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.	—
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show flow interface** *type number* **Example:** `Device# show flow interface GigabitEthernet 0/0/0`	Displays the status of Flexible NetFlow (enabled or disabled) on the specified interface.
Step 8	**show flow monitor name** *monitor-name* **cache format record** **Example:** `Device# show flow monitor name FLOW_MONITOR-1 cache format record`	Displays the status, statistics, and flow data in the cache for the specified flow monitor.

Configuration Examples for Flexible NetFlow Layer 2 Fields

Example: Configuring Flexible NetFlow for Monitoring MAC and VLAN Statistics

The following example shows how to configure Flexible NetFlow for monitoring MAC and VLAN statistics.

This example starts in global configuration mode.

```
!
 flow record LAYER-2-FIELDS-1
 match ipv4 source address
 match ipv4 destination address
 match datalink dot1q vlan output
 match datalink mac source address input
 match datalink mac source address output
 match datalink mac destination address input
 match flow direction
!
 exit
!
!
flow monitor FLOW-MONITOR-4
 record LAYER-2-FIELDS-1
 exit
!
ip cef
```

```
!
interface GigabitEthernet0/0/1
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input
!
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

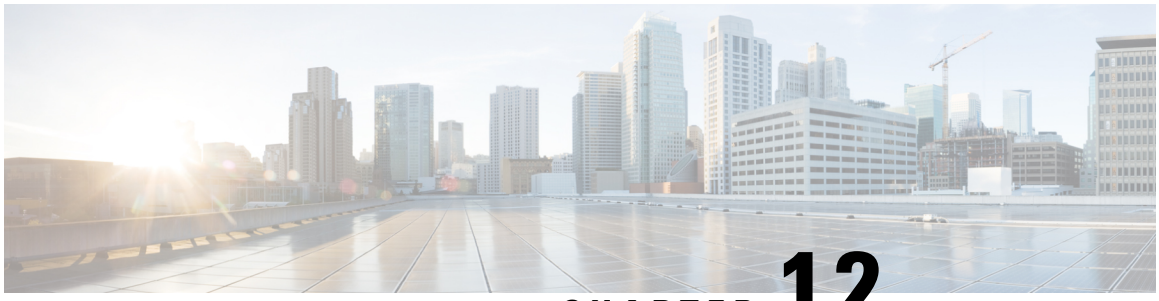
Feature Information for Flexible NetFlow - Layer 2 Fields

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

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

Table 26: Feature Information for Flexible NetFlow - Layer 2 Fields

Feature Name	Releases	Feature Information
Flexible NetFlow - Layer 2 Fields	12.2(33)SRE 12.4(22)T Cisco IOS XE Release 3.2SE	Enables collecting statistics for Layer 2 fields such as MAC addresses and virtual LAN (VLAN) IDs from traffic. Support for this feature was added for Cisco 7200 and 7300 Network Processing Engine (NPE) series routers in Cisco IOS Release 12.2(33)SRE. The following commands were introduced or modified: **collect datalink dot1q vlan**, **collect datalink mac**, **match datalink dot1q vlan**, **match datalink mac**.



CHAPTER 12

Flexible Netflow - Ingress VRF Support

The Flexible Netflow - Ingress VRF Support feature enables collecting the virtual routing and forwarding (VRF) ID from incoming packets on a router by applying an input flow monitor having a flow record that collects the VRF ID as a key or a nonkey field.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible NetFlow Ingress VRF Support

Flexible NetFlow—Ingress VRF Support Overview

This feature enables collecting the virtual routing and forwarding (VRF) ID from incoming packets on a router by applying an input flow monitor having a flow record that collects the VRF ID as a key or a nonkey field.

How to Configure Flexible NetFlow Ingress VRF Support

Creating a Customized Flow Record

Perform this task to configure a customized flow record.

Customized flow records are used to analyze traffic data for a specific purpose. A customized flow record must have at least one **match** criterion for use as the key field and typically has at least one **collect** criterion for use as a nonkey field.

There are hundreds of possible permutations of customized flow records. This task shows the steps that are used to create one of the possible permutations. Modify the steps in this task as appropriate to create a customized flow record for your requirements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **description** *description*
5. **match** {**ip** | **ipv6**} {**destination** | **source**} **address**
6. Repeat Step 5 as required to configure additional key fields for the record.
7. **collect interface** {**input** | **output**}
8. Repeat the above step as required to configure additional nonkey fields for the record.
9. **end**
10. **show flow record** *record-name*
11. **show running-config flow record** *record-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-RECORD-1`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. • This command also allows you to modify an existing flow record.

	Command or Action	Purpose
Step 4	**description** *description* **Example:** `Device(config-flow-record)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow record.
Step 5	**match** {**ip** | **ipv6**} {**destination** | **source**} **address** **Example:** `Device(config-flow-record)# match ipv4 destination address`	Configures a key field for the flow record. **Note** This example configures the IPv4 destination address as a key field for the record. For information about the other key fields available for the **match ipv4** command, and the other **match** commands that are available to configure key fields.
Step 6	Repeat Step 5 as required to configure additional key fields for the record.	—
Step 7	**collect interface** {**input** | **output**} **Example:**	Configures the input interface as a nonkey field for the record. **Note** This example configures the input interface as a nonkey field for the record.
Step 8	Repeat the above step as required to configure additional nonkey fields for the record.	—
Step 9	**end** **Example:** `Device(config-flow-record)# end`	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 10	**show flow record** *record-name* **Example:** `Device# show flow record FLOW_RECORD-1`	(Optional) Displays the current status of the specified flow record.
Step 11	**show running-config flow record** *record-name* **Example:** `Device# show running-config flow record FLOW_RECORD-1`	(Optional) Displays the configuration of the specified flow record.

Creating a Customized Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be one of the predefined formats or a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record instead of using one of the Flexible NetFlow predefined records, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}}
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **statistics packet protocol**
9. **statistics packet size**
10. **exporter** *exporter-name*
11. **end**
12. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]]
13. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:**	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode.

	Command or Action	Purpose
	`(config)# flow monitor FLOW-MONITOR-1`	• This command also allows you to modify an existing flow monitor.
Step 4	**description** *description* **Example:** `(config-flow-monitor)# description Used for basic ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}} **Example:**	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. The values for the keywords associated with the **timeout** keyword have no effect when the cache type is set to **immediate**.
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	**statistics packet protocol** **Example:** `(config-flow-monitor)# statistics packet protocol`	(Optional) Enables the collection of protocol distribution statistics for Flexible NetFlow monitors.
Step 9	**statistics packet size** **Example:** `(config-flow-monitor)# statistics packet size`	(Optional) Enables the collection of size distribution statistics for Flexible NetFlow monitors.
Step 10	**exporter** *exporter-name* **Example:** `(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.
Step 11	**end** **Example:** `(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 12	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] **Example:**	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.

	Command or Action	Purpose
	`# show flow monitor FLOW-MONITOR-2 cache`	
Step 13	**show running-config flow monitor** *monitor-name* **Example:** `# show running-config flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Applying a Flow Monitor to an Interface

Before it can be activated, a flow monitor must be applied to at least one interface. Perform this required task to activate a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
5. Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.
6. **end**
7. **show flow interface** *type number*
8. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.
Step 4	{**ip**	**ipv6**} **flow monitor** *monitor-name* {**input**	**output**} **Example:**	Activates a flow monitor that was created previously by assigning it to the interface to analyze traffic.

	Command or Action	Purpose
	`Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	
Step 5	Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.	—
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show flow interface** *type number* **Example:** `Device# show flow interface GigabitEthernet 0/0/0`	Displays the status of Flexible NetFlow (enabled or disabled) on the specified interface.
Step 8	**show flow monitor name** *monitor-name* **cache format record** **Example:** `Device# show flow monitor name FLOW_MONITOR-1 cache format record`	Displays the status, statistics, and flow data in the cache for the specified flow monitor.

Configuration Examples for Flexible NetFlow Ingress VRF Support

Example: Configuring Flexible NetFlow for Ingress VRF Support

The following example configures the collection of the virtual routing and forwarding (VRF) ID from incoming packets on a router by applying an input flow monitor having a flow record that collects the VRF ID as a key field.

This example starts in global configuration mode.

```
!
flow record rm_1
match routing vrf input
match ipv4 source address
match ipv4 destination address
collect interface input
collect interface output
collect counter packets
!
flow monitor mm_1
record rm_1
!
interface GigabitEthernet 0/0/0
```

```
ip vrf forwarding green
ip address 172.16.2.2 255.255.255.252
ip flow monitor mm_1 input
!
end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

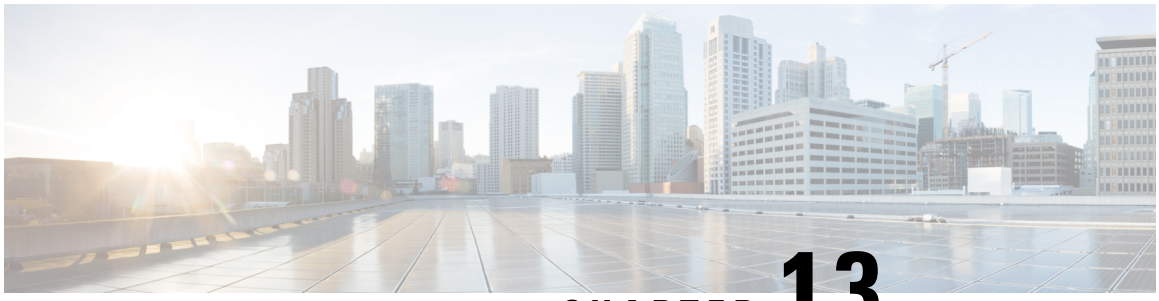
Feature Information for Flexible NetFlow—Ingress VRF Support

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

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

Table 27: Feature Information for Flexible NetFlow—Ingress VRF Support

Feature Name	Releases	Feature Information
Flexible NetFlow--Ingress VRF Support	12.2(33)SRE 12.2(50)SY 15.0(1)M 15.0(1)SY 15.0(1)SY1	Enables collecting the virtual routing and forwarding (VRF) ID from incoming packets on a router by applying an input flow monitor having a flow record that collects the VRF ID as a key or a nonkey field. Support for this feature was added for Cisco 7200 and 7300 Network Processing Engine (NPE) series routers in Cisco IOS Release 12.2(33)SRE. The following commands were introduced or modified: **collect routing, match routing, option (Flexible NetFlow), show flow monitor.**



CHAPTER 13

Flexible NetFlow NBAR Application Recognition Overview

NBAR enables creation of different flows for each application seen between any two IP hosts by applying a flow monitor having a flow record that collects the application name as a key or a nonkey field.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible NetFlow NBAR Application Recognition

Flexible NetFlow NBAR Application Recognition Overview

If the predefined Flexible NetFlow records are not suitable for your traffic requirements, you can create a user-defined (custom) record using the Flexible NetFlow **collect** and **match** commands. Before you can create a customized record, you must decide the criteria that you are going to use for the key and nonkey fields.

If you want to create a customized record for detecting network attacks, you must include the appropriate key and nonkey fields in the record to ensure that the router creates the flows and captures the data that you need

to analyze the attack and respond to it. Flexible NetFlow uses Network-based Application recognition (NBAR) to create different flows for each application seen between any two IP hosts by applying a flow monitor having a flow record that collects the application name as a key field.

How to Configure Flexible NetFlow NBAR Application Recognition

Creating a Customized Flow Record

Perform this task to configure a customized flow record.

Customized flow records are used to analyze traffic data for a specific purpose. A customized flow record must have at least one **match** criterion for use as the key field and typically has at least one **collect** criterion for use as a nonkey field.

There are hundreds of possible permutations of customized flow records. This task shows the steps that are used to create one of the possible permutations. Modify the steps in this task as appropriate to create a customized flow record for your requirements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **description** *description*
5. **match** {**ip** | **ipv6**} {**destination** | **source**} **address**
6. Repeat Step 5 as required to configure additional key fields for the record.
7. **collect interface** {**input** | **output**}
8. Repeat the above step as required to configure additional nonkey fields for the record.
9. **end**
10. **show flow record** *record-name*
11. **show running-config flow record** *record-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-RECORD-1`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. • This command also allows you to modify an existing flow record.
Step 4	**description** *description* **Example:** `Device(config-flow-record)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow record.
Step 5	**match** {**ip** | **ipv6**} {**destination** | **source**} **address** **Example:** `Device(config-flow-record)# match ipv4 destination address`	Configures a key field for the flow record. **Note** This example configures the IPv4 destination address as a key field for the record. For information about the other key fields available for the **match ipv4** command, and the other **match** commands that are available to configure key fields.
Step 6	Repeat Step 5 as required to configure additional key fields for the record.	—
Step 7	**collect interface** {**input** | **output**} **Example:**	Configures the input interface as a nonkey field for the record. **Note** This example configures the input interface as a nonkey field for the record.
Step 8	Repeat the above step as required to configure additional nonkey fields for the record.	—
Step 9	**end** **Example:** `Device(config-flow-record)# end`	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 10	**show flow record** *record-name* **Example:** `Device# show flow record FLOW_RECORD-1`	(Optional) Displays the current status of the specified flow record.
Step 11	**show running-config flow record** *record-name* **Example:** `Device# show running-config flow record FLOW_RECORD-1`	(Optional) Displays the configuration of the specified flow record.

Creating a Customized Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be one of the predefined formats or a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record instead of using one of the Flexible NetFlow predefined records, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.

Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}}
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **statistics packet protocol**
9. **statistics packet size**
10. **exporter** *exporter-name*
11. **end**
12. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]]
13. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**flow monitor** *monitor-name* **Example:** `(config)# flow monitor FLOW-MONITOR-1`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. • This command also allows you to modify an existing flow monitor.
Step 4	**description** *description* **Example:** `(config-flow-monitor)# description Used for basic ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name* | **netflow-original** | **netflow** {**ipv4** | **ipv6**} *record* [**peer**]} **Example:** `(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | {**immediate** | **normal** | **permanent**}} **Example:**	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. The values for the keywords associated with the **timeout** keyword have no effect when the cache type is set to **immediate**.
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	**statistics packet protocol** **Example:** `(config-flow-monitor)# statistics packet protocol`	(Optional) Enables the collection of protocol distribution statistics for Flexible NetFlow monitors.
Step 9	**statistics packet size** **Example:** `(config-flow-monitor)# statistics packet size`	(Optional) Enables the collection of size distribution statistics for Flexible NetFlow monitors.
Step 10	**exporter** *exporter-name* **Example:** `(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.
Step 11	**end** **Example:** `(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 12	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] **Example:** `# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status and statistics for a Flexible NetFlow flow monitor.
Step 13	**show running-config flow monitor** *monitor-name* **Example:** `# show running-config flow monitor FLOW_MONITOR-1`	(Optional) Displays the configuration of the specified flow monitor.

Applying a Flow Monitor to an Interface

Before it can be activated, a flow monitor must be applied to at least one interface. Perform this required task to activate a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. {**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**}
5. Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.
6. **end**
7. **show flow interface** *type number*
8. **show flow monitor name** *monitor-name* **cache format record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Specifies an interface and enters interface configuration mode.

	Command or Action	Purpose
Step 4	{**ip** | **ipv6**} **flow monitor** *monitor-name* {**input** | **output**} **Example:** `Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	Activates a flow monitor that was created previously by assigning it to the interface to analyze traffic.
Step 5	Repeat Steps 3 and 4 to activate a flow monitor on any other interfaces in the device over which you want to monitor traffic.	—
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show flow interface** *type number* **Example:** `Device# show flow interface GigabitEthernet 0/0/0`	Displays the status of Flexible NetFlow (enabled or disabled) on the specified interface.
Step 8	**show flow monitor name** *monitor-name* **cache format record** **Example:** `Device# show flow monitor name FLOW_MONITOR-1 cache format record`	Displays the status, statistics, and flow data in the cache for the specified flow monitor.

Configuration Examples for Flexible NetFlow NBAR Application Recognition

Example: Configuring Flexible NetFlow for Network-Based Application Recognition

```
!
flow record rm_1
match application name
match ipv4 source address
match ipv4 destination address
collect interface input
collect interface output
collect counter packets
!
flow monitor mm_1
record rm_1
!
interface FastEthernet0/0
```

```
ip address 172.16.2.2 255.255.255.0
ip flow monitor mm_1 input
!
end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

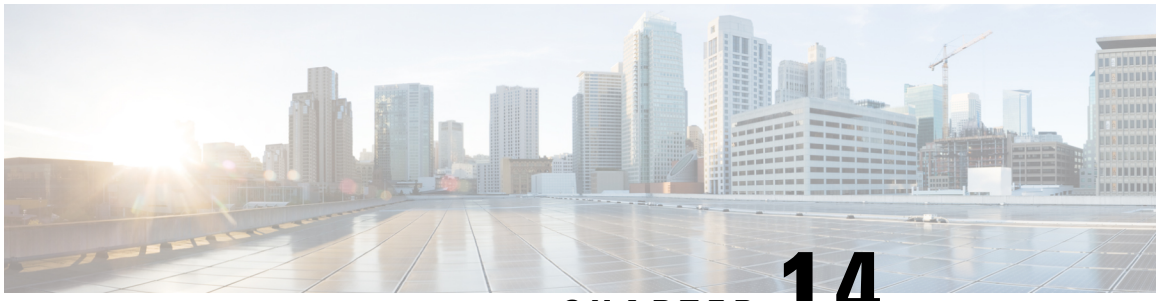
Feature Information for Flexible NetFlow NBAR Application Recognition

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

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

Table 28: Feature Information for Flexible NetFlow NBAR Application Recognition

Feature Name	Releases	Feature Information
Flexible NetFlow--NBAR Application Recognition		Network-based Application recognition (NBAR) enables creation of different flows for each application seen between any two IP hosts by applying a flow monitor having a flow record that collects the application name as a key or a nonkey field. The following commands were introduced or modified: **collect application name**, **match application name**, **option** (Flexible NetFlow), **show flow monitor**.



CHAPTER 14

Flexible NetFlow IPFIX Export Format

The Flexible NetFlow IPFIX Export Format feature enables sending export packets using the IPFIX export protocol. The export of extracted fields from NBAR is only supported over IPFIX.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible NetFlow IPFIX Export Format

Flexible NetFlow IPFIX Export Format Overview

IPFIX is an IETF standard based on NetFlow v9.

The Flexible NetFlow IPFIX Export Format feature enables sending export packets using the IPFIX export protocol. The export of extracted fields from NBAR is only supported over IPFIX.

How to Configure Flexible NetFlow IPFIX Export Format

Configuring the Flow Exporter

Perform this required task to configure the flow exporter.

Note Each flow exporter supports only one destination.

You can export to a destination using either an IPv4 or IPv6 address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **dscp** *dscp*
7. **source** *interface-type interface-number*
8. **output-features**
9. **template data timeout** *seconds*
10. **transport udp** *udp-port*
11. **ttl** *seconds*
12. **end**
13. **show flow exporter** *exporter-name*
14. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:**	Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode.

	Command or Action	Purpose
	`Device(config)# flow exporter EXPORTER-1`	• This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to the datacenter`	(Optional) Configures a description to the exporter that will appear in the configuration and the display of the **show flow exporter** command.
Step 5	**destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the IP address or hostname of the destination system for the exporter. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**dscp** *dscp* **Example:** `Device(config-flow-exporter)# dscp 63`	(Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter. • The range for the *dscp* argument is from 0 to 63. Default: 0.
Step 7	**source** *interface-type interface-number* **Example:** `Device(config-flow-exporter)# source ethernet 0/0`	(Optional) Specifies the local interface from which the exporter will use the IP address as the source IP address for exported datagrams.
Step 8	**output-features** **Example:** `Device(config-flow-exporter)# output-features`	(Optional) Enables sending export packets using quality of service (QoS) and encryption.
Step 9	**template data timeout** *seconds* **Example:** `Device(config-flow-exporter)# template data timeout 120`	(Optional) Configures resending of templates based on a timeout. • The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours).
Step 10	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 650`	Specifies the UDP port on which the destination system is listening for exported datagrams. • The range for the *udp-port* argument is from 1 to 65536.
Step 11	**ttl** *seconds* **Example:** `Device(config-flow-exporter)# ttl 15`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. • The range for the *seconds* argument is from 1 to 255.

	Command or Action	Purpose
Step 12	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flow exporter configuration mode and returns to privileged EXEC mode.
Step 13	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.
Step 14	**show running-config flow exporter** *exporter-name* **Example:** `Device# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Configuration Examples for Flexible NetFlow IPFIX Export Format

Example: Configuring Flexible NetFlow IPFIX Export Format

The following example shows how to configure IPFIX export format for Flexible NetFlow.

This sample starts in global configuration mode:

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
 export-protocol ipfix
 transport udp 90
 exit
!
flow monitor FLOW-MONITOR-1
 record netflow ipv4 original-input
 exporter EXPORTER-1
!
ip cef
!
interface Ethernet 0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input
!
```

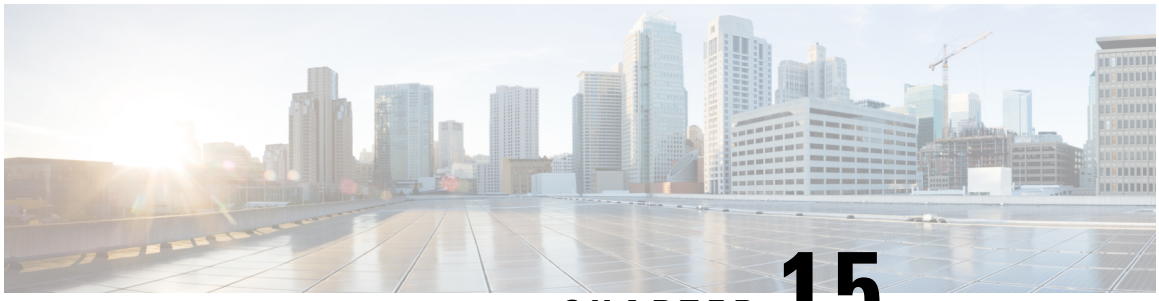
Feature Information for Flexible NetFlow: IPFIX Export Format

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

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

Table 29: Feature Information for Flexible NetFlow : IPFIX Export Format

Feature Name	Releases	Feature Information
Flexible NetFlow: IPFIX Export Format	15.2(4)M Cisco IOS XE Release 3.7S 15.2(1)SY	Enables sending export packets using the IPFIX export protocol. The export of extracted fields from NBAR is only supported over IPFIX. Support for this feature was added for Cisco ASR 1000 Series Aggregation Services routers in Cisco IOS XE Release 3.7S. The following command was introduced: **export-protocol**.



CHAPTER 15

Flexible Netflow Export to an IPv6 Address

The Export to an IPv6 Address feature enables Flexible NetFlow to export data to a destination using an IPv6 address.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Information About Flexible Netflow Export to an IPv6 Address

Flexible Netflow Export to an IPv6 Address Overview

This feature enables Flexible NetFlow to export data to a destination using an IPv6 address.

How to Configure Flexible Netflow Export to an IPv6 Address

Configuring the Flow Exporter

Perform this required task to configure the flow exporter.

Note Each flow exporter supports only one destination.

You can export to a destination using either an IPv4 or IPv6 address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **dscp** *dscp*
7. **source** *interface-type interface-number*
8. **output-features**
9. **template data timeout** *seconds*
10. **transport udp** *udp-port*
11. **ttl** *seconds*
12. **end**
13. **show flow exporter** *exporter-name*
14. **show running-config flow exporter** *exporter-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter EXPORTER-1`	Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode. • This command also allows you to modify an existing flow exporter.
Step 4	**description** *description* **Example:** `Device(config-flow-exporter)# description Exports to the datacenter`	(Optional) Configures a description to the exporter that will appear in the configuration and the display of the **show flow exporter** command.

	Command or Action	Purpose
Step 5	**destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*] **Example:** `Device(config-flow-exporter)# destination 172.16.10.2`	Specifies the IP address or hostname of the destination system for the exporter. **Note** You can export to a destination using either an IPv4 or IPv6 address.
Step 6	**dscp** *dscp* **Example:** `Device(config-flow-exporter)# dscp 63`	(Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter. • The range for the *dscp* argument is from 0 to 63. Default: 0.
Step 7	**source** *interface-type interface-number* **Example:** `Device(config-flow-exporter)# source ethernet 0/0`	(Optional) Specifies the local interface from which the exporter will use the IP address as the source IP address for exported datagrams.
Step 8	**output-features** **Example:** `Device(config-flow-exporter)# output-features`	(Optional) Enables sending export packets using quality of service (QoS) and encryption.
Step 9	**template data timeout** *seconds* **Example:** `Device(config-flow-exporter)# template data timeout 120`	(Optional) Configures resending of templates based on a timeout. • The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours).
Step 10	**transport udp** *udp-port* **Example:** `Device(config-flow-exporter)# transport udp 650`	Specifies the UDP port on which the destination system is listening for exported datagrams. • The range for the *udp-port* argument is from 1 to 65536.
Step 11	**ttl** *seconds* **Example:** `Device(config-flow-exporter)# ttl 15`	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. • The range for the *seconds* argument is from 1 to 255.
Step 12	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flow exporter configuration mode and returns to privileged EXEC mode.
Step 13	**show flow exporter** *exporter-name* **Example:** `Device# show flow exporter FLOW_EXPORTER-1`	(Optional) Displays the current status of the specified flow exporter.

	Command or Action	Purpose
Step 14	**show running-config flow exporter** *exporter-name* **Example:** `Device# show running-config flow exporter FLOW_EXPORTER-1`	(Optional) Displays the configuration of the specified flow exporter.

Configuration Examples for Flexible Netflow Export to an IPv6 Address

Example: Configuring Multiple Export Destinations

The following example shows how to configure multiple export destinations for Flexible NetFlow for IPv4 or IPv6 traffic. This sample starts in global configuration mode:

The following example shows how to configure multiple export destinations for Flexible NetFlow for IPv4:

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
 transport udp 90
 exit
!
flow exporter EXPORTER-2
 destination 172.16.10.3
 transport udp 90
 exit
!
flow record v4_r1
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long


flow monitor FLOW-MONITOR-1
 record v4_r1
 exporter EXPORTER-2
 exporter EXPORTER-1
!

ip cef
!
interface GigabitEthernet1/0/0
 ip address 172.16.6.2 255.255.255.0
 ip flow monitor FLOW-MONITOR-1 input

!
```

The following example shows how to configure multiple export destinations for Flexible NetFlow for IPv6:

```
!
flow exporter EXPORTER-1
 destination 172.16.10.2
 transport udp 90
 exit
!
flow exporter EXPORTER-2
 destination 172.16.10.3
 transport udp 90
 exit
!

flow record v6_r1
match ipv6 traffic-class
match ipv6 protocol
match ipv6 source address
match ipv6 destination address
match transport source-port
match transport destination-port
collect counter bytes long
collect counter packets long
!


!
flow monitor FLOW-MONITOR-2
 record v6_r1
 exporter EXPORTER-2
 exporter EXPORTER-1
!
ip cef
!
interface GigabitEthernet1/0/0
 ipv6 address 2001:DB8:2:ABCD::2/48
 ipv6 flow monitor FLOW-MONITOR-2 input
!
```

The following display output shows that the flow monitor is exporting data to the two exporters:

```
Device# show flow monitor FLOW-MONITOR-1
Flow Monitor FLOW-MONITOR-1:
  Description:       User defined
  Flow Record:       v4_r1
  Flow Exporter:     EXPORTER-1
                     EXPORTER-2
  Cache:
    Type:              normal (Platform cache)
    Status:            allocated
    Size:              4096 entries / 311316 bytes
    Inactive Timeout:  15 secs
    Active Timeout:    1800 secs
    Update Timeout:    1800 secs
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

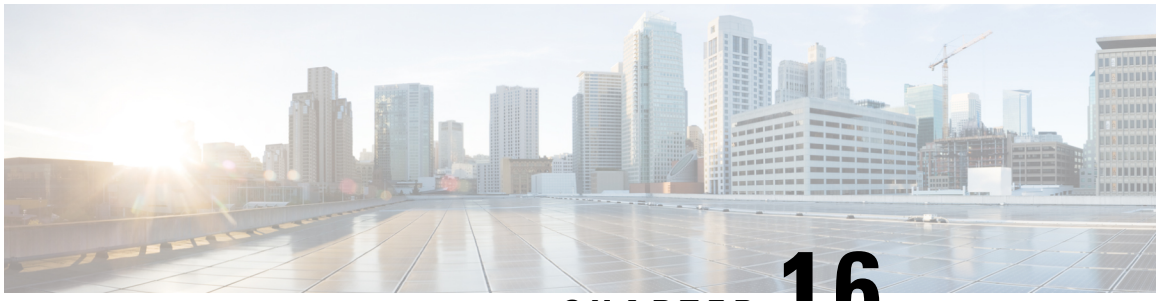
Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 16

Flexible NetFlow: Integration with MQC

This module describes the Flexible NetFlow: Integration with MQC feature. Flexible NetFlow supports the creation of flow caches with specific flow information tailored to the various services used in the network. A flow group allows customers to target and see only specific types of traffic and therefore increases the scalability of NetFlow while giving the customer the ability to target and isolate specific types of network behavior or IP applications. Flow groups use a Modular Quality of Service CLI (MQC) filtering and classification applied to a NetFlow cache.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Prerequisites for Flexible NetFlow: Integration with MQC

- You are familiar with the information in the *Flexible NetFlow Configuration Guide*.
- You are familiar with the MQC information in the "Applying QoS Features Using the MQC" module.
- The networking device must be running a Cisco IOS release that supports Flexible NetFlow.

IPv4 Traffic

- The networking device must be configured for IPv4 routing.

- One of the following must be enabled on your router and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding or distributed Cisco Express Forwarding.

IPv6 Traffic

- The networking device must be configured for IPv6 routing.
- One of the following must be enabled on your router and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding IPv6 or distributed Cisco Express Forwarding IPv6.

Information About Flexible NetFlow Integration with MQC

Flexible NetFlow: Integration with MQC Overview

Flexible NetFlow supports the creation of flow caches with specific flow information tailored to the various services used in the network. A flow group allows customers to target and see only specific types of traffic and therefore increases the scalability of NetFlow while giving the customer the ability to target and isolate specific types of network behavior or IP applications. Flow groups use MQC filtering and classification applied to a NetFlow cache.The Flexible NetFlow: Integration with MQC feature integrates FNF with MQC traffic selection and classification technology to support per-flow accounting for a subset of traffic received or sent on an interface or subinterface.

How to Configure Flexible NetFlow Integration with MQC

Configuring Flexible NetFlow: Integration with MQC

To configure the Flexible NetFlow: Integration with MQC feature, perform the following required task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **match ipv4 source address**
5. **match ipv4 destination address**
6. **match application name**
7. **exit**
8. **flow monitor** *monitor-name*
9. **description** *description*
10. **exit**
11. **sampler** *sampler-name*
12. **exit**
13. **class-map** *class-map-name*
14. **exit**

15. **policy-map** *policy-map-name*
16. **class** *class-map-name*
17. **flow monitor** *monitor-name* **sampler** *sampler-name*
18. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-MONITOR-1`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. • This command also allows you to modify an existing flow record.
Step 4	**match ipv4 source address** **Example:** `Router(config-flow-record)# match ipv4 source address`	Identifies the IPv4 source address as a match criterion.
Step 5	**match ipv4 destination address** **Example:** `Router(config-flow-record)# match ipv4 destination address`	Identifies the IPv4 destination address as a match criterion.
Step 6	**match application name** **Example:** `Router(config-flow-record)# match application name`	Identifies the application name as a match criterion.
Step 7	**exit** **Example:** `Router(config-flow-record)# exit`	Exits Flexible NetFlow flow record configuration mode and returns to global configuration mode.
Step 8	**flow monitor** *monitor-name* **Example:**	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode.

	Command or Action	Purpose
	`Device(config)# flow monitor FLOW-MONITOR-1`	• This command also allows you to modify an existing flow monitor.
Step 9	**description** *description* **Example:** `Device(config-flow-monitor)# description Used for basic traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 10	**exit** **Example:** `Router(config-flow-monitor)# exit`	Exits Flexible NetFlow flow monitor configuration mode and returns to global configuration mode.
Step 11	**sampler** *sampler-name* **Example:** `Device(config)# sampler sm_1`	Creates a flow sampler and enters Flexible NetFlow sampler configuration mode. • This command also allows you to modify an existing flow sampler.
Step 12	**exit** **Example:** `Router(config-sampler)# exit`	Exits Flexible NetFlow sampler configuration mode and returns to global configuration mode.
Step 13	**class-map** *class-map-name* **Example:** `Router(config)# class-map cmap`	Creates a class to be used with a class map and enters QoS class-map configuration mode.
Step 14	**exit** **Example:** `Router(config-cmap)# exit`	Exits QoS class-map configuration mode and returns to global configuration mode.
Step 15	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map pmap`	Specifies the name of the policy map to be created and enters QoS policy-map configuration mode.
Step 16	**class** *class-map-name* **Example:** `Router(config-pmap)# class cmap`	Specifies the name of the class of the policy to be created and enters QoS policy-map class configuration mode.
Step 17	**flow monitor** *monitor-name* **sampler** *sampler-name* **Example:**	Configures the flow monitor and sampler as a MQC policy map class action.

	Command or Action	Purpose
	`Device(config-pmap-c)# flow monitor FLOW-MONITOR-1 sampler sm_1`	
Step 18	**end** **Example:** `Device(config-pmap-c)# end`	Exits QoS policy-map class configuration mode and returns to privileged EXEC mode.

Configuration Examples for Flexible NetFlow Integration with MQC

Example: Configuring Flexible NetFlow: Integration with MQC

The following example displays a sample configuration for the functionality provided by the Flexible NetFlow: Integration with MQC feature:

```
enable
configure terminal
flow record rm_1
 match ipv4 source address
 match ipv4 destination address
 match application name
flow monitor mm_1
sampler sm_1
class-map cmap
policy-map pmap
 class cmap
  flow monitor mm_1 sampler sm_1
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Flexible NetFlow conceptual information and configuration tasks	*Flexible NetFlow Configuration Guide*
Flexible NetFlow commands	*Cisco IOS Flexible NetFlow Command Reference*

Standards/RFCs

Standard	Title
No new or modified standards/RFCs are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Flexible NetFlow: Integration with MQC

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

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

Table 30: Feature Information for Flexible NetFlow: Integration with MQC

Feature Name	Releases	Feature Information
Flexible NetFlow: Integration with MQC	15.2(4)M 15.3(1)T	Flexible NetFlow supports the creation of flow caches with specific flow information tailored to the various services used in the network. A flow group allows customers to target and see only specific types of traffic and therefore increases the scalability of NetFlow while giving the customer the ability to target and isolate specific types of network behavior or IP applications. Flow groups use MQC filtering and classification applied to a NetFlow cache. The following commands were introduced or modified: **flow monitor**.