

Flexible NetFlow—IPv6 Unicast Flows

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

- Information About Flexible NetFlow IPv6 Unicast Flows, on page 1
- How to Configure Flexible NetFlow IPv6 Unicast Flows, on page 1
- Configuration Examples for Flexible NetFlow IPv6 Unicast Flows, on page 12

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 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. **match flow cts** {**source** | **destination**} **group-tag**
8.
9. Repeat the above step as required to configure additional nonkey fields for the record.
10. **end**
11. **show flow record** *record-name*
12. **show running-config flow record** *record-name*

DETAILED STEPS

Procedure

	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`	**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	**match flow cts** {**source** | **destination**} **group-tag** **Example:**	**Note** This example configures the CTS source group tag and destination group tag as a key field for the record. For

	Command or Action	Purpose

	`Device(config-flow-record)# match flow cts source group-tag`
	
	`Device(config-flow-record)# match flow cts destination group-tag`
		information about the other key fields available for the **match ipv4**/**ipv6** command, and the other **match** commands that are available to configure key fields.
		
		Note
		• Ingress:
		• In an incoming packet, if a header is present, SGT will reflect the same value as the header. If no value is present, it will show zero.
		• The DGT value will not depend on the ingress port SGACL configuration.
		• Egress:
		• If either propagate SGT or CTS is disabled on the egress interface, then SGT will be zero.
		• In an outgoing packet, if SGACL configuration that corresponds to the (SGT, DGT) exists, DGT will be non-zero.
		• If SGACL is disabled on the egress port/VLAN or if global SGACL enforcement is disabled, then DGT will be zero

Step 8	**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 9	Repeat the above step as required to configure additional nonkey fields for the record.	—
Step 10	**end** **Example:** `Device(config-flow-record)# end`	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 11	**show flow record** *record-name* **Example:** `Device# show flow record FLOW_RECORD-1`	(Optional) Displays the current status of the specified flow record.
Step 12	**show running-config flow record** *record-name* **Example:**	(Optional) Displays the configuration of the specified flow record.

	Command or Action	Purpose
	`Device# show running-config flow record FLOW_RECORD-1`	

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

Procedure

	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 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.
Step 11	**ttl** *seconds* **Example:**	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter.

	Command or Action	Purpose
	`Device(config-flow-exporter)# ttl 15`	• 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 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 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, 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*}
6. **cache** {**timeout** {**active**} *seconds* | { **normal** }
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.
8. **exporter** *exporter-name*

9. **end**
10. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]]]
11. **show running-config flow monitor** *monitor-name*

DETAILED STEPS

Procedure

	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 ipv4 traffic analysis`	(Optional) Creates a description for the flow monitor.
Step 5	**record** {*record-name*} **Example:** `Device(config-flow-monitor)# record FLOW-RECORD-1`	Specifies the record for the flow monitor.
Step 6	**cache** {**timeout** {**active**} *seconds* | { **normal** } **Example:**	
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter EXPORTER-1`	(Optional) Specifies the name of an exporter that was created previously.

	Command or Action	Purpose
Step 9	**end** **Example:** `Device(config-flow-monitor)# end`	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 10	**show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]]] **Example:** `Device# show flow monitor FLOW-MONITOR-2 cache`	(Optional) Displays the status for a Flexible NetFlow flow monitor.
Step 11	**show running-config flow monitor** *monitor-name* **Example:** `Device# 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.

While running the **ip flow monitor** command for the first interface to enable FNF monitor, you may see the following warning message displaying a GLOBAL memory allocation failure. This log is triggered by enabling FNF monitoring with a large cache size.

```
Jul  4 01:45:00.255: %CPPEXMEM-3-NOMEM: F0/0: cpp_cp_svr: QFP: 0, GLOBAL memory allocation
 of 90120448 bytes by FNF failed
Jul  4 01:45:00.258: %CPPEXMEM-3-TOPUSER: F0/0: cpp_cp_svr: QFP: 0, Top User: CPR STILE
EXMEM GRAPH, Allocations: 877, Type: GLOBAL
Jul  4 01:45:00.258: %CPPEXMEM-3-TOPUSER: F0/0: cpp_cp_svr: QFP: 0, Top User: SBC, Bytes
Allocated: 53850112, Type: GLOBAL
```

The warning message does not necessarily indicate a flow monitor application failure. The warning message can indicate internal steps that FNF uses for applying memory from the EXMEM infrastructure.

To ensure that the FNF monitor is enabled successfully, use the **show flow monitor** *monitor-name* command to check **Status** (**allocated** or **not allocated**) of a flow monitor. For more information, see Displaying the Current Status of 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

Procedure

	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.

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

Procedure

	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:**	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 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:** `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
!
```