Configuring NetFlow

Use this chapter to configure NetFlow to characterize IP traffic based on its source, destination, timing, and application information, to assess network availability and performance.
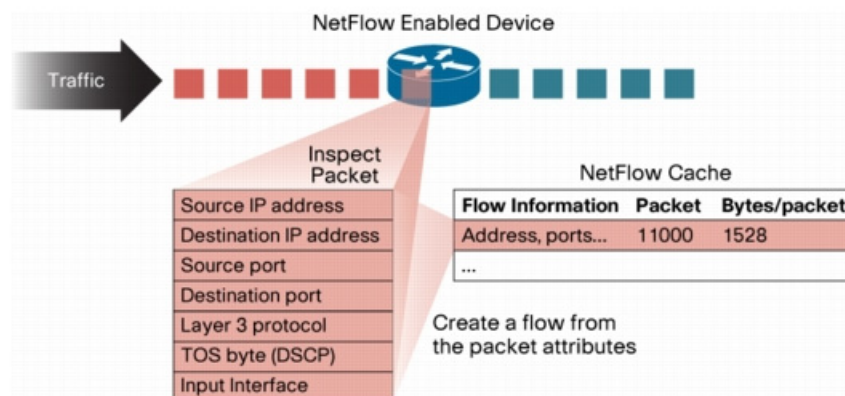
Information About NetFlow

NetFlow lets you evaluate IP traffic and understand how and where it flows. NetFlow gathers data that can be used in accounting, network monitoring, and network planning.

What is a Flow

A flow is a one-directional stream of packets that arrives on a source interface (or subinterface), matching a set of criteria. All packets with the same source/destination IP address, source/destination ports, protocol interface and class of service are grouped into a flow and then packets and bytes are tallied. This condenses a large amount of network information into a database called the NetFlow cache.

Figure 11-1   Creating a Flow in the NetFlow Cache

You create a flow by defining the criteria it gathers. Flows are stored in the NetFlow cache.

Flow information tells you the following:

  • Source address tells you who is originating the traffic.

- Destination address tells who is receiving the traffic.
- Ports characterize the application using the traffic.
- Class of service examines the priority of the traffic.
- The device interface tells how traffic is being used by the network device.
- Tallied packets and bytes show the amount of traffic.

Flow Record Definition

A flow record defines the information that NetFlow gathers, such as packets in the flow and the types of counters gathered per flow. You can define new flow records or use the predefined Cisco Nexus 1000V flow record.

To create a record, see the .

The following table describes the criteria defined in a flow record.

Flow record criteria	Description

Match	Defines what information is matched for collection in the flow record.
	
	 • **ip**: Data collected in the flow record matches one of the following IP options:
	   – protocol
	   – tos (type of service)
	
	 • **ipv4**: Data collected in the flow record matches one of the following ipv4 address options:
	   – source address
	   – destination adress
	
	 • **transport**: Data collected in the flow record matches one of the following transport options:
	   – destination port
	   – source port

Collect	Defines how the flow record collects information.
	
	 • **counter**: Collects Flow Record information in one of the following formats:
	   – bytes: collected in 32-bit counters unless the long 64-bit counter is specified.
	   – packets: collected in 32-bit counters unless the long 64-bit counter is specified.
	
	 • **timestamp sys-uptime**: Collects the system up time for the first or last packet in the flow.
	
	 • **transport tcp flags**: Collects the TCP transport layer flags for the packets in the flow.

Predefined Flow Records

Cisco Nexus 1000V includes the following pre-defined flow records.

Example 11-1   Cisco Nexus 1000V Predefined Flow Record: Netflow-Original

```
n1000v# show flow record netflow-original
Flow record netflow-original:
    Description: Traditional IPv4 input NetFlow with origin ASs
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 source address
        match ipv4 destination address
        match ip protocol
        match ip tos
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect routing source as
        collect routing destination as
        collect routing next-hop address ipv4
        collect transport tcp flags
        collect counter bytes
        collect counter packets
        collect timestamp sys-uptime first
        collect timestamp sys-uptime last
n1000v#
```

> **Note**   Although the following lines appear in the output of the **show flow record** command, the commands they are based on are not currently supported in Cisco Nexus 1000V. The use of these commands has no affect on the configuration.
>
> ```
> collect routing source as
> collect routing destination as
> collect routing next-hop address ipv4
> ```

Example 11-2   Cisco Nexus 1000V Predefined Flow Record: Netflow IPv4 Original-Input

```
n1000v# show flow record netflow ipv4 original-input
Flow record ipv4 original-input:
    Description: Traditional IPv4 input NetFlow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 source address
        match ipv4 destination address
        match ip protocol
        match ip tos
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect routing source as
        collect routing destination as
        collect routing next-hop address ipv4
```

```
            collect transport tcp flags
            collect counter bytes
            collect counter packets
            collect timestamp sys-uptime first
            collect timestamp sys-uptime last
n1000v#
```

Example 11-3   Cisco Nexus 1000V Predefined Flow Record: Netflow IPv4 Original-Output

```
switch# show flow record netflow ipv4 original-output
Flow record ipv4 original-output:
    Description: Traditional IPv4 output NetFlow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 source address
        match ipv4 destination address
        match ip protocol
        match ip tos
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect routing source as
        collect routing destination as
        collect routing next-hop address ipv4
        collect transport tcp flags
        collect counter bytes
        collect counter packets
        collect timestamp sys-uptime first
        collect timestamp sys-uptime last
switch#
```

Example 11-4   Cisco Nexus 1000V Predefined Flow Record: Netflow IPv4 Protocol-Port

```
switch# show flow record netflow ipv4 protocol-port
Flow record ipv4 protocol-port:
    Description: Protocol and Ports aggregation scheme
    No. of users: 0
    Template ID: 0
    Fields:
        match ip protocol
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect counter bytes
        collect counter packets
        collect timestamp sys-uptime first
        collect timestamp sys-uptime last
switch#
```

Accessing NetFlow Data

There are two primary methods used to access NetFlow data:

- Command Line Interface (CLI), page 11-5
- NetFlow Collector, page 11-6

Command Line Interface (CLI)

To view what is happening in your network now, use the CLI. To see a list of available show commands, see the "Verifying the NetFlow Configuration" section on page 11-18.

The CLI uses the following tools to capture and export flow records to the Netflow Collector:

- Flow Monitor, page 11-5
- Flow Exporter, page 11-5

Flow Monitor

A flow monitor creates an association between the following NetFlow components:

- a flow record—consisting of matching and collection criteria
- a flow exporter—consisting of the export criteria

This flow monitor association enables a set, consisting of a record and an exporter, to be defined once and re-used many times. Multiple flow monitors can be created for different needs. A flow monitor is applied to a specific interface in a specific direction.

See the "Defining a Flow Monitor" procedure on page 11-15, and "Assigning a Flow Monitor to an Interface" procedure on page 11-17.

Flow Exporter

Use the flow exporter to define where and when the flow records are sent from the cache to the reporting server, called the NetFlow Collector.

An exporter definition includes the following.

- Destination IP address
- Source interface
- UDP port number (where the collector is listening)
- Export format

> **Note** NetFlow export packets use the IP address assigned to the source interface. If the source interface does not have an IP address assigned to it, the exporter will be inactive.

See the "Defining a Flow Exporter" procedure on page 11-13.

Export Formats

Cisco Nexus 1000V supports the NetFlow Version 9 export format.

> **Note** Cisco Nexus 1000V supports UDP as the transport protocol for exporting data to up to two exporters per monitor.

NetFlow Collector

You can export NetFlow from the Cisco Nexus 1000V NetFlow cache to a reporting server called the NetFlow Collector. The NetFlow Collector assembles the exported flows and combines them to produce reports used for traffic and security analysis. NetFlow export, unlike SNMP polling, pushes information periodically to the NetFlow reporting collector. The NetFlow cache is constantly filling with flows. Cisco Nexus 1000V searches the cache for flows that have terminated or expired and exports them to the NetFlow collector server. Flows are terminated when the network communication has ended, that is, when a packet contains the TCP FIN flag.

The following steps implement NetFlow data reporting:

- NetFlow records are configured to define the information that NetFlow gathers.
- Netflow monitor is configured to capture flow records to the NetFlow cache.
- NetFlow export is configured to send flows to the collector.
- Cisco Nexus 1000V searches the NetFlow cache for flows that have terminated and exports them to the NetFlow collector server.
- Flows are bundled together based on space availability in the UDP export packet or based on export timer.
- The NetFlow collector software creates real-time or historical reports from the data.

Exporting Flows to the NetFlow Collector Server

Timers determine when a flow is exported to the NetFlow Collector Server.

A flow is ready for export when one of the following occurs:

- The flow is inactive for a certain time during which no new packets are received for the flow.
- The flow has lived longer than the active timer, for example, a long FTP download.
- A TCP flag indicates the flow is terminated. That is, a FIN or RST flag is present.
- The flow cache is full and some flows must be aged out to make room for new flows.

Figure 11-2   Exporting Flows to the NetFlow Collector Server
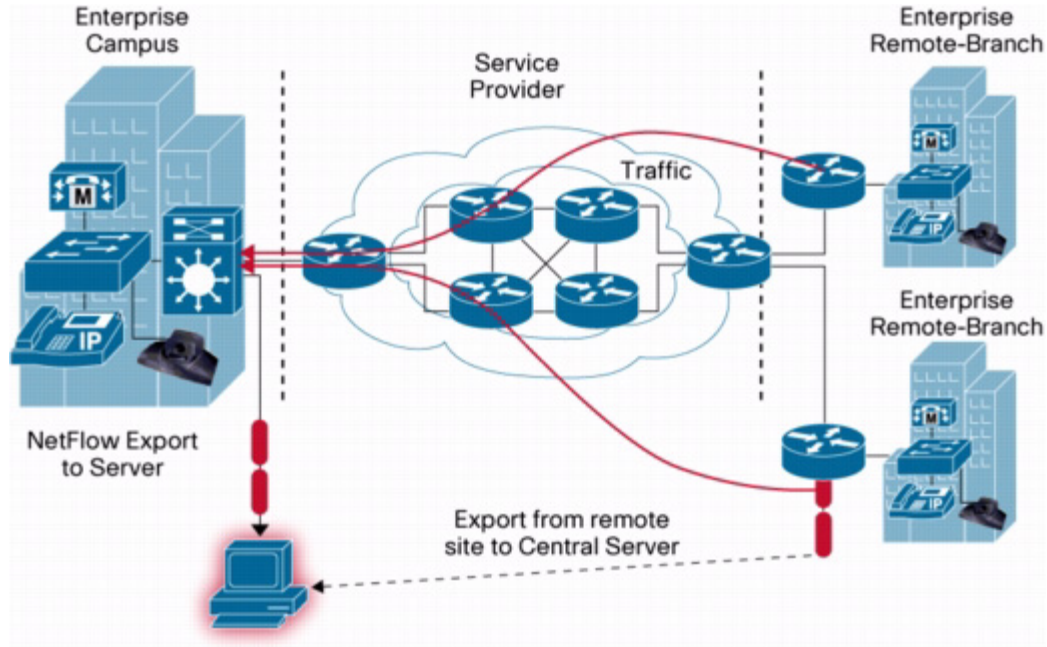
What NetFlow Data Looks Like

The following figure shows an example of NetFlow data.

Figure 11-3   NetFlow Cache Example
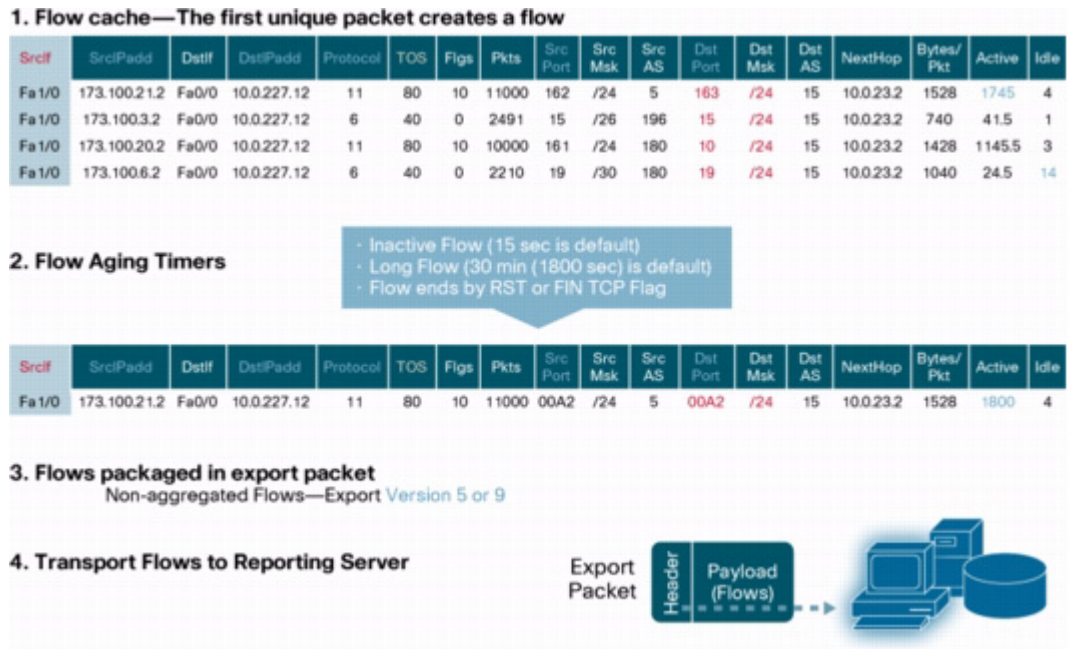

1. Flow cache—The first unique packet creates a flow

Srcif	SrcIPadd	Dstif	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	162	/24	5	163	/24	15	10.0.23.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.0.23.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	161	/24	180	10	/24	15	10.0.23.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.0.23.2	1040	24.5	14

2. Flow Aging Timers

· Inactive Flow (15 sec is default)
· Long Flow (30 min (1800 sec) is default)
· Flow ends by RST or FIN TCP Flag

Srcif	SrcIPadd	Dstif	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1800	4

3. Flows packaged in export packet
    Non-aggregated Flows—Export Version 5 or 9

4. Transport Flows to Reporting Server

Export Packet   Header   Payload (Flows)

Network Analysis Module

You can also use the Cisco Network Analysis Module (NAM) to monitor NetFlow data sources. NAM enables traffic analysis views and reports such as hosts, applications, conversations, VLAN, and QoS.

To use NAM for monitoring the Cisco Nexus 1000V NetFlow data sources see the *Cisco Nexus 1010 Network Analysis Module Installation and Configuration Note, 4.2*.

High Availability

Cisco Nexus 1000V supports stateful restarts for NetFlow. After a reboot or supervisor switchover, Cisco Nexus 1000V applies the running configuration.

Prerequisites for NetFlow

- You must be aware of resource requirements since NetFlow consumes additional memory and CPU resources.

- Memory and CPU resources are provided by the VEM hosting the flow monitor interface. Resources are limited by the number of CPU cores present on the VEM.

Configuration Guidelines and Limitations

NetFlow has the following configuration guidelines and limitations:

- If a source interface is not configured, the NetFlow exporter will remain disabled.

- In Cisco Nexus 1000V, Mgmt0 interface is configured by default as the source interface for an exporter. You can change the source interface if needed.

- Cisco Nexus 1000V includes the following predefined flow records that can be used instead of configuring a new one. For more information, see the "Flow Record Definition" section on page 11-2:

  - netflow-original
    Cisco Nexus 1000V predefined traditional IPv4 input NetFlow with origin ASs

    > ✎
    > **Note**   The routing-related fields in this predefined flow record are ignored.

  - netflow ipv4 original-input
    Cisco Nexus 1000V predefined traditional IPv4 input NetFlow

  - netflow ipv4 original-output
    Cisco Nexus 1000V predefined traditional IPv4 output NetFlow

  - netflow ipv4 protocol-port
    Cisco Nexus 1000V predefined `protocol and ports aggregation scheme`

- Up to 256 NetFlow interfaces are allowed per DVS.

- Up to 32 NetFlow interfaces are allowed per host

- A maximum of one flow monitor per interface per direction is allowed.

- Up to 8 flow monitors are allowed per VEM.

- Up to 2 flow exporters are permitted per monitor.

- Up to 32 NetFlow Policies are allowed per DVS.

- Up to 8 NetFlow Policies are allowed per host.

- NetFlow is not supported on port channels.

Default Settings

Table 11-1 lists the default settings for NetFlow parameters.

Table 11-1   Default NetFlow Parameters

Parameters	Default
NetFlow version	9
source interface	mgmt0
match	direction and interface (incoming/outgoing)
flow monitor active timeout	1800
flow monitor inactive timeout	15
flow monitor cache size	4096
flow exporter UDP port **transport udp** command	9995
DSCP	default/best-effort (0)
VRF	default

Configuring NetFlow

The following flow chart is designed to guide you through the netflow configuration process. After completing each procedure, return to the flow chart to make sure you complete all required procedures in the correct sequence.

Flow Chart: Configuring NetFlow
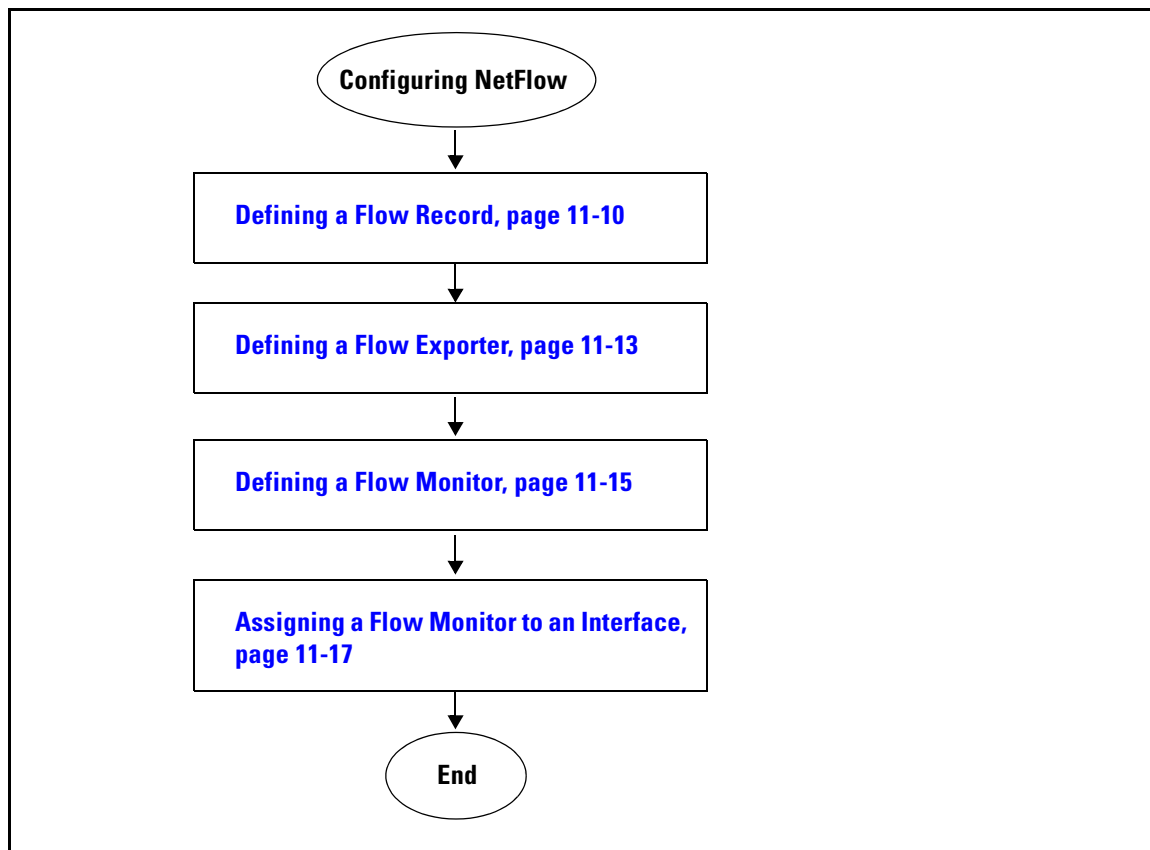
Defining a Flow Record

Use this procedure to create a flow record.

> **Note** Optionally, you can use the Cisco Nexus 1000V pre-defined record shown in the "Flow Record Definition" section on page 11-2. See the "Defining a Flow Monitor" section on page 11-15 to apply a pre-defined record to a flow monitor.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You know which of the options you want this flow record to match.
- You know which options you want this flow record to collect.

  For more information, see the "Flow Record Definition" section on page 11-2.

> **Note** Although the following lines appear in the output of the **show flow record** command, the commands they are based on are not currently supported in Cisco Nexus 1000V. The use of these commands has no affect on the configuration.

```
collect routing source as
collect routing destination as
collect routing next-hop address ipv4
```

SUMMARY STEPS

1. **config t**

2. **flow record** *name*

3. **description** *string*

4. **match {ip {protocol | tos} | ipv4 {destination address | source address} | transport {destination-port | source-port}}**

5. **collect {counter {bytes [long] | packets [long]} | timestamp sys-uptime | transport tcp flags}**

6. **show flow record** [*name*]

7. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	*config t*
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	**flow record** *name*
	
	Example:
	`n1000v(config)# flow record RecordTest`
	`n1000v(config-flow-record)#`
		Creates a Flow Record by name, and places you in the CLI Flow Record Configuration mode for that specific record.

Step 3	**description** *string* **Example:** `n1000v(config-flow-record)# description Ipv4Flow`	(Optional) Adds a description of up to 63 characters to this Flow Record and saves it in the running configuration.

Step 4	**match {ip{protocol| tos}|ipv4{destination address|source address}|transport {destination-port|source-port}}**
	
	Example:
	`n1000v(config-flow-record)# match ipv4 destination address`
		Defines the Flow Record to match one of the following and saves it in the running configuration.
		
		• **ip**: Matches one of the following IP options:
		  – protocol
		  – tos (type of service)
		• **ipv4**: Matches one of the following ipv4 address options:
		  – source address
		  – destination adress
		• **transport**: Matches one of the following transport options:
		  – destination port
		  – source port

	Command	Purpose
Step 5	**collect** {**counter** {**bytes** [**long**] | **packets** [**long**]} | **timestamp sys-uptime** | **transport tcp flags**} **Example:** `n1000v(config-flow-record)# collect counter packets`	Specifies a collection option to define the information to collect in the Flow Record and saves it in the running configuration. • **counter**: Collects Flow Record information in one of the following formats: – bytes: collected in 32-bit counters unless the long 64-bit counter is specified. – packets: collected in 32-bit counters unless the long 64-bit counter is specified. • **timestamp sys-uptime**: Collects the system up time for the first or last packet in the flow. • **transport tcp flags**: Collects the TCP transport layer flags for the packets in the flow.
Step 6	**show flow record** [*name*] **Example:** `n1000v(config-flow-exporter)# show flow record RecordTest`	(Optional) Displays information about Flow Records.
Step 7	**copy running-config startup-config** **Example:** `n1000v(config-flow-exporter)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example for creating a flow record:

```
n1000v# config t
n1000v(config)# flow record RecordTest
n1000v(config-flow-record)# description Ipv4flow
n1000v(config-flow-record)# match ipv4 destination address
n1000v(config-flow-record)# collect counter packets
n1000v(config-flow-record)# show flow record RecordTest
Flow record RecordTest:
    Description: Ipv4flow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 destination address
        match interface input
        match interface output
        match flow direction
        collect counter packets
n1000v(config-flow-record)#
```

Defining a Flow Exporter

Use this procedure to create a Flow Exporter defining where and how Flow Records are exported to the NetFlow Collector Server.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- A maximum of two flow exporters per monitor are permitted.
- You know destination IP address of the NetFlow Collector Server.
- You know the source interface that Flow Records are sent from.
- You know the transport UDP that the Collector is listening on.
- Export format version 9 is the version supported.

SUMMARY STEPS

1. **config t**
2. **flow exporter** *name*
3. **description** *string*
4. **destination** {*ipv4-address* | *ipv6-address*}
5. **dscp** *value*
6. **source mgmt** *interface_number*
7. **transport udp** *port-number*
8. **version 9**
9. **option {exporter-stats | interface-table} timeout** *seconds*
10. **template data timeout** *seconds*
11. **show flow exporter** [*name*]
12. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**flow exporter** *name*
	
	Example:
	`n1000v(config)# flow exporter ExportTest`
	`n1000v(config-flow-exporter)#`
		Creates a Flow Exporter, saves it in the running configuration, and then places you in CLI Flow Exporter Configuration mode.

Step 3	**description** *string*
	
	Example:
	`n1000v(config-flow-exporter)#`
	`description ExportV9`
		Adds a description of up to 63 characters to this Flow Exporter and saves it in the running configuration.

	Command	Purpose
Step 4	**destination** {*ipv4-address*	*ipv6-address*} **Example:** `n1000v(config-flow-exporter)# destination 192.0.2.1`	Specifies the IP address of the destination interface for this Flow Exporter and saves it in the running configuration.
Step 5	**dscp** *value* **Example:** `n1000v(config-flow-exporter)# dscp 0`	Specifies the differentiated services codepoint value for this Flow Exporter, between 0 and 63, and saves it in the running configuration.
Step 6	**source mgmt** *interface_number* **Example:** `n1000v(config-flow-exporter)# source mgmt 0`	Specifies the interface and its number, from which the Flow Records are sent to the NetFlow Collector Server, and saves it in the running configuration.
Step 7	**transport udp** *port-number* **Example:** `n1000v(config-flow-exporter)# transport udp 200`	Specifies the destination UDP port, between 0 and 65535, used to reach the NetFlow collecton, and saves it in the running configuration.

Step 8	**version** {**9**}
	
	Example:
	`n1000v(config-flow-exporter)# version 9`
	`n1000v(config-flow-exporter-version-9)#`
		Specifies NetFlow export version 9, saves it in the running configuration, and places you into the export version 9 configuration mode.

Step 9	**option** {**exporter-stats**	**interface-table**	**sampler-table**} **timeout** *value* **Example:** `n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200`	Specifies one of the following version 9 exporter resend timers and its value, between 1 and 86400 seconds, and saves it in the running configuration. • **exporter-stats** • **interface-table** • **sampler-tabl**e
Step 10	**template data timeout** *seconds* **Example:** `n1000v(config-flow-exporter-version-9)# template data timeout 1200`	Sets the template data resend timer and its value, between 1 and 86400 seconds, and saves it in the running configuration.
Step 11	**show flow exporter** [*name*] **Example:** `n1000v(config-flow-exporter)# show flow exporter`	(Optional) Displays information about the Flow Exporter.
Step 12	**copy running-config startup-config** **Example:** `n1000v(config-flow-exporter)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example of creating a flow exporter:

```
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
```

```
n1000v(config-flow-exporter-version-9)# show flow exporter ExportTest
Flow exporter ExportTest:
    Description: ExportHamilton
    Destination: 192.0.2.1
    VRF: default (1)
    Destination UDP Port 200
    Source Interface Mgmt0
    DSCP 2
    Export Version 9
        Exporter-stats timeout 1200 seconds
        Data template timeout 1200 seconds
    Exporter Statistics
        Number of Flow Records Exported 0
        Number of Templates Exported 0
        Number of Export Packets Sent 0
        Number of Export Bytes Sent 0
        Number of Destination Unreachable Events 0
        Number of No Buffer Events 0
        Number of Packets Dropped (No Route to Host) 0
        Number of Packets Dropped (other) 0
        Number of Packets Dropped (LC to RP Error) 0
        Number of Packets Dropped (Output Drops) 1
        Time statistics were last cleared: Never
n1000v(config-flow-exporter-version-9)#
```

Defining a Flow Monitor

Use this procedure to create a Flow Monitor and associate a Flow Record and a Flow Exporter to it.

BEFORE YOU BEGIN

- A maximum of one flow monitor per interface per direction is permitted.

- You know the name of an existing Flow Exporter to associate with this flow monitor.

- You know the name of an existing Flow Record to associate with this flow monitor. You can use either a flow record you previously created, or one of the following Cisco Nexus 1000V predefined flow records:

  - netflow-original

  - netflow ipv4 original-input

  - netflow ipv4 original-output

  - netflow ipv4 protocol-port

  For more information about Flow Records, see the "Flow Record Definition" section on page 11-2

SUMMARY STEPS

1. **config t**

2. **flow monitor** *name*

3. **description** *string*

4. **exporter** *name*

5. **record** *name*

6. **timeout** {**active** *value* | **inactive** *value*}

7. **cache** {**size** *value*}

8. **show flow monitor** [*name*]

9. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	**flow monitor** *name*
	
	Example:
	`n1000v(config)# flow monitor MonitorTest`
	`n1000v(config-flow-monitor)#`
		Creates a flow monitor, by name, saves it in the running configuration, and then places you in the CLI Flow Monitor Configuration mode,

Step 3	**description** *string* **Example:** `n1000v(config-flow-monitor)# description Ipv4Monitor`	(Optional) For the specified flow monitor, adds a descriptive string, of up to 63 alphanumeric characters, and saves it in the running configuration.
Step 4	**exporter** *name* **Example:** `n1000v(config-flow-monitor)# exporter Exportv9`	For the specified flow monitor, adds an existing flow exporter and saves it in the running configuration.

Step 5	**record** {*name* | **netflow** {**ipv4**}}
	
	Example using Cisco Nexus 1000V pre-defined record:
	`n1000v(config-flow-monitor)# record netflow-original`
	
	Example using user-defined record:
	`n1000v(config-flow-monitor)# record RecordTest`
		For the specified flow monitor, adds an existing flow record and saves it in the running configuration.
		
		• **name**: The name of a flow record you have previously created, or the name of a Cisco provided pre-defined flow record.
		
		• **netflow**: Traditional NetFlow collection schemes
		
		– **ipv4**: Traditional IPv4 NetFlow collection schemes

Step 6	**timeout** {**active** *value* | **inactive** *value*} **Example:** `n1000v(config-flow-monitor)# timeout inactive 600`	(Optional) For the specified flow monitor, specifies an aging timer and its value for aging entries from the cache, and saves them in the running configuration. • **active**: Active, or long, timeout. Allowable values are from 60 to 4092 seconds. Default is 1800. • **inactive**: Inactive or normal timeout. Allowable values are from 15 to 4092 seconds. Default is 15.
Step 7	**cache** {**size** *value*} **Example:** `n1000v(config-flow-monitor)# cache size 15000`	(Optional) For the specified flow monitor, specifies the cache size, from 256 to 16384, entries, and saves it in the running configuration. Default is 4096. **Note** This option is used to limit the impact of the monitor cache on memory and performance.

	Command	Purpose

Step 8	**show flow monitor** [*name*]
	
	Example:
	`n1000v(config-flow-monitor)# show flow`
	`monitor Monitor Test`
		(Optional) Displays information about existing flow monitors.

Step 9	**copy running-config startup-config**
	
	Example:
	`n1000v(config-flow-monitor)# copy`
	`running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example of creating a flow exporter:

```
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record RecordTest
n1000v(config-flow-monitor)# cache size 15000
n1000v(config-flow-monitor)# timeout inactive 600
n1000v(config-flow-monitor)# show flow monitor MonitorTest
Flow Monitor monitortest:
    Use count: 0
    Inactive timeout: 600
    Active timeout: 1800
    Cache Size: 15000
n1000v(config-flow-monitor)#
```

Assigning a Flow Monitor to an Interface

Use this procedure to assign a flow monitor to an interface.

BEFORE YOU BEGIN

- You know the name of the flow monitor you want to use for the interface.
- You know the interface type and its number.

SUMMARY STEPS

1. **config t**
2. **interface** *interface-type interface-number*
3. **ip flow monitor** *name* {**input** | **output**}
4. **show flow** *interface-type interface-number*
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	**interface** *interface-type interface-number*
	
	Example:
	`n1000v(config)# interface veth 2`
	`n1000v(config-if)#`
		Places you in the CLI Interface Configuration mode for the specified interface.

Step 3	**ip flow monitor** *name* {**input** | **output**}
	
	Example:
	`n1000v(config-if)# ip flow monitor`
	`MonitorTest output`
		For the specified interface, assigns a flow monitor for input or output packets and saves it in the running configuration.

Step 4	**show flow** *interface-type interface-number*
	
	Example:
	`n1000v(config-if# show flow interface`
	`veth 2`
		(Optional) For the specified interface, displays the NetFlow configuration.

Step 5	**copy running-config startup-config**
	
	Example:
	`n1000v(config-if)# copy running-config`
	`startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example showing how to assign a flow monitor to an interface:

```
n1000v(config)# interface veth 2
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2
Interface veth 2:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#
```

Verifying the NetFlow Configuration

To verify the NetFlow configuration, use the commands in Table 11-2:

Table 11-2   Verifying the NetFlow Configuration

Command	Purpose
show flow exporter [*name*]	Displays information about NetFlow flow exporter maps. See Example 11-5 on page 11-19.
show flow interface [*interface-type number*]	Displays information about NetFlow interfaces. See Example 11-6 on page 11-20.

Table 11-2   Verifying the NetFlow Configuration (continued)

Command	Purpose
show flow monitor [*name* [**cache module** *number* |**statistics module** *number*]]	Displays information about NetFlow flow monitors.
	Note The **show flow monitor cache** command differs from the **show flow monitor statistics** command in that the cache command also displays cache entries. Since each processor has its own cache, all output of these commands is based on the number of processors on the server (also called module or host). When more than one processor is involved in processing packets for a single flow, then the same flow appears for each processor.
	See the following examples:
	• Example 11-7Show flow monitor, page 11-20
	• Example 11-8Show flow monitor cache module, page 11-20
	• Example 11-9Show flow monitor statistics module, page 11-21
show flow record [*name*]	Displays information about NetFlow flow records.

Example 11-5   Show flow exporter

```
n1000v(config-flow-exporter-version-9)# show flow exporter ExportTest
Flow exporter ExportTest:
    Description: ExportHamilton
    Destination: 192.0.2.1
    VRF: default (1)
    Destination UDP Port 200
    Source Interface 2
    DSCP 2
    Export Version 9
        Exporter-stats timeout 1200 seconds
        Data template timeout 1200 seconds
    Exporter Statistics
        Number of Flow Records Exported 0
        Number of Templates Exported 0
        Number of Export Packets Sent 0
        Number of Export Bytes Sent 0
        Number of Destination Unreachable Events 0
        Number of No Buffer Events 0
        Number of Packets Dropped (No Route to Host) 0
        Number of Packets Dropped (other) 0
        Number of Packets Dropped (LC to RP Error) 0
        Number of Packets Dropped (Output Drops) 1
        Time statistics were last cleared: Never
n1000v(config-flow-exporter-version-9)#
```

Example 11-6   Show flow interface

```
n1000v(config-if)# show flow interface VEth2
Interface veth2:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#
```

Example 11-7   Show flow monitor

```
n1000v(config)# show flow monitor
Flow Monitor MonitorTest:
    Description: Ipv4Monitor
    Use count: 1
    Flow Record: test
    Flow Exporter: ExportTest
    Inactive timeout: 15
    Active timeout: 1800
    Cache Size: 15000
Flow Monitor MonitorIpv4:
    Description: exit
    Use count: 70
    Flow Record: RecordTest
    Flow Exporter: ExportIpv4
    Inactive timeout: 15
    Active timeout: 1800
    Cache Size: 4096
n1000v(config)#
```

Example 11-8   Show flow monitor cache module

```
n1000v# show flow monitor test_mon cache module 5
Cache type:                     Normal
Cache size (per-processor):     4096
High Watermark:                 2
Flows added:                    102
Flows aged:                     099
    - Active timeout            0
    - Inactive timeout          099
    - Event aged                0
    - Watermark aged            0
    - Emergency aged            0
    - Permanent                 0
    - Immediate aged            0
    - Fast aged                 0

Cache entries on Processor0
    - Active Flows:             2
    - Free Flows:              4094
```

IPV4 SRC ADDR	IPV4 DST ADDR	IP PROT	INTF INPUT	INTF OUTPUT	FLOW DIRN
0.0.0.0	255.255.255.255	17	Veth1		Input
7.192.192.10	7.192.192.2	1	Veth1	Eth5/2	Input

```
Cache entries on Processor1
    - Active Flows:             0
    - Free Flows:              4096

Cache entries on Processor2
    - Active Flows:             1
    - Free Flows:              4095
```

```
  IPV4 SRC ADDR     IPV4 DST ADDR  IP PROT               INTF INPUT              INTF OUTPUT  FLOW DIRN
=============== =============== ======= ===================== ===================== =========
   7.192.192.10     7.192.192.1  1                           Veth1                   Eth5/2    Input

Cache entries on Processor3
    - Active Flows:              0
    - Free Flows:            4096

Cache entries on Processor4
    - Active Flows:              0
    - Free Flows:            4096

Cache entries on Processor5
    - Active Flows:              0
    - Free Flows:            4096

Cache entries on Processor6
    - Active Flows:              0
    - Free Flows:            4096

Cache entries on Processor7
    - Active Flows:              0
    - Free Flows:            4096
```

Example 11-9   Show flow monitor statistics module

```
NX-1000v# show flow monitor test_mon statistics module 5
Cache type:                   Normal
Cache size (per-processor):   4096
High Watermark:               2
Flows added:                  105
Flows aged:                   103
    - Active timeout          0
    - Inactive timeout        103
    - Event aged              0
    - Watermark aged          0
    - Emergency aged          0
    - Permanent               0
    - Immediate aged          0
    - Fast aged               0

Cache entries on Processor0
    - Active Flows:              0
    - Free Flows:            4096

Cache entries on Processor1
    - Active Flows:              1
    - Free Flows:            4095

Cache entries on Processor2
    - Active Flows:              1
    - Free Flows:            4095

Cache entries on Processor3
    - Active Flows:              0
    - Free Flows:            4096

Cache entries on Processor4
    - Active Flows:              0
    - Free Flows:            4096

Cache entries on Processor5
```

```
            - Active Flows:             0
            - Free Flows:            4096

Cache entries on Processor6
            - Active Flows:             0
            - Free Flows:            4096

Cache entries on Processor7
            - Active Flows:             0
            - Free Flows:            4096
```

Example 11-10 Show flow record

```
n1000v(config-flow-record)# show flow record RecordTest
Flow record RecordTest:
    Description: Ipv4flow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 destination address
        match interface input
        match interface output
        match flow direction
        collect counter packets
n1000v(config-flow-record)#
```

Configuration Example for NetFlow

The following example shows how to configure a flow monitor using a new flow record and applying it to an interface.

```
n1000v# config t
n1000v(config)# flow record RecordTest
n1000v(config-flow-record)# description Ipv4flow
n1000v(config-flow-record)# match ipv4 destination address
n1000v(config-flow-record)# collect counter packets
n1000v(config-flow-record)# exit
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
n1000v(config-flow-exporter-version-9)# exit
n1000v(config-flow-exporter)# exit
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record RecordTest
n1000v(config-flow-monitor)# exit
n1000v(config)# interface veth 2/1
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2
Interface veth 2:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#
```

The following example shows how to configure flow monitor using a predefined record and applying it to an interface.

```
n1000v# config t
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
n1000v(config-flow-exporter-version-9)# exit
n1000v(config-flow-exporter)# exit
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record netflow-original
n1000v(config-flow-monitor)# exit
n1000v(config)# interface veth 2/1
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2
Interface veth 2:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#
```

Additional References

For additional information related to implementing NetFlow, see the following sections:

- Related Documents, page 11-23
- Standards, page 11-24

Related Documents

Related Topic	Document Title
Cisco NetFlow Overview	http://cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html
Complete command syntax, command mode, command history, defaults, usage guidelines, and examples for all Cisco Nexus 1000V commands.	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for NetFlow

This section provides the NetFlow feature release history.

Feature Name	Releases	Feature Information
NAM support for NetFlow data sources	4.0(4)SV1(3)	NAM support for NetFlow data sources was added.
NetFlow	4.0(4)SV1(1)	NetFlow was introduced.