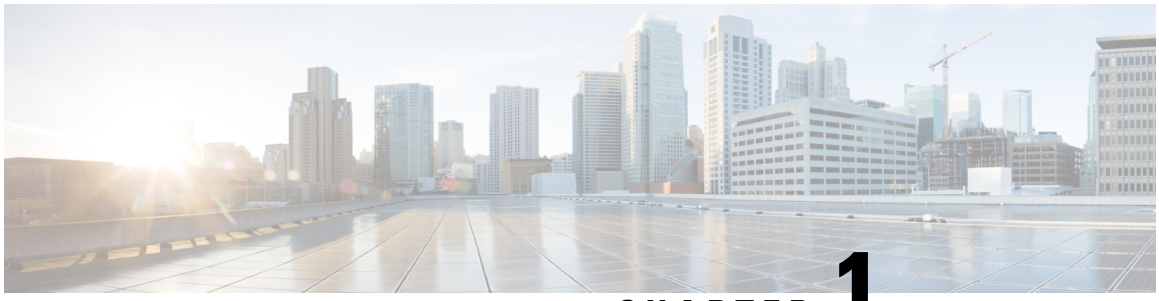
# Media Monitoring Configuration Guide, Cisco IOS XE Gibraltar 16.12.x


**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

# Read Me First

### Important Information about Cisco IOS XE 16

Effective Cisco IOS XE Release 3.7.0E for Catalyst Switching and Cisco IOS XE Release 3.17S (for Access and Edge Routing) the two releases evolve (merge) into a single version of converged release—the Cisco IOS XE 16—providing one release covering the extensive range of access and edge products in the Switching and Routing portfolio.

### Feature Information

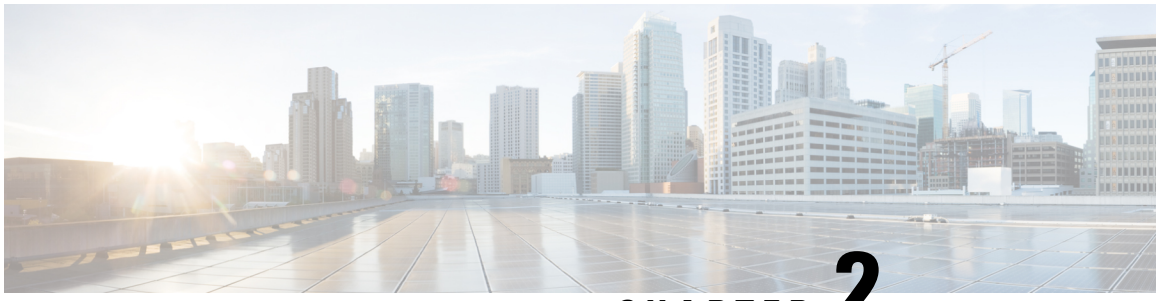
Use Cisco Feature Navigator to find information about feature support, platform support, and Cisco software image support. An account on Cisco.com is not required.

### Related References

- Cisco IOS Command References, All Releases

### Obtaining Documentation and Submitting a Service Request

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.



CHAPTER 2

# Configuring Cisco Mediatrace

This chapter contains information about and instructions for configuring Cisco Mediatrace.

Cisco Mediatrace enables you to isolate and troubleshoot network degradation problems for data streams. Although it can be used to monitor any type of flow, it is primary used with video flows. It can also be used for non-flow related monitoring along a media flow path.



# 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 Configuring Cisco Mediatrace

## Overview of Cisco Mediatrace

**Note** Mediatrace is no longer supported on M&T train. For performance monitoring, see Configuring Cisco Performance Monitor, on page 37 chapter.

Cisco Mediatrace helps to isolate and troubleshoot network degradation problems by enabling a network administrator to discover an IP flow's path, dynamically enable monitoring capabilities on the nodes along the path, and collect information on a hop-by-hop basis. This information includes, among other things, flow statistics, and utilization information for incoming and outgoing interfaces, CPUs, and memory, as well as any changes to IP routes or the Cisco Mediatrace monitoring state.

This information can be retrieved in either of two ways:

- By issuing an exec command to perform an on-demand collection of statistics from the hops along a media flow. During this one-shot operation, the hops along the media flow are discovered and shown to you, along with a set of other specified information.
- By configuring Cisco Mediatrace to start a recurring monitoring session at a specific time and on specific days. The session can be configured to specify which metrics to collect, and how frequently they are collected. The hops along the path are automatically discovered as part of the operation.

After collecting the metrics you specified, you can view a report on the metrics.

Cisco Mediatrace is part of the Cisco Medianet family of products. For more information about the design, configuration, and troubleshooting of Mediatrace when used in conjunction with other Cisco products, including a Quick Start Guide and Deployment Guide, see the Cisco Medianet Knowledge Base Portal, located at http://www.cisco.com/web/solutions/medianet/knowledgebase/index.html.

## Metrics That You Can Collect Using Cisco Mediatrace

You can collect the following categories of metrics using Mediatrace:

- Common Metrics for Each Responder
- System Metrics: TCP Profile
- System Metrics: RTP Profile
- System Metrics: INTF Profile
- System Metrics: CPU Profile
- System Metrics: MEMORY Profile
- App-Health Metrics: MEDIATRACE-HEALTH Profile
- Metrics for the Mediatrace Request Summary from Initiator

The individual metrics under each of these categories are listed the appropriate section below.

### Metics for Mediatrace Request Summary from Initiator

- Request Timestamp
- Request Status
- Number of Hops Responded
- Number of Hops with Valid Data
- Number of Hops with Error
- Number of hops with no data record

- Last Route Change Timestamp
- Route Index

### Common Metrics for Each Responder

- Metrics Collection Status
- Reachability address
- Ingress Interface
- Egress Interface
- Mediatrace IP TTL
- Hostname
- Mediatrace Hop Count

### Perf-Monitor Metrics: TCP Profile

- Flow Sampling Start Timestamp
- Loss of measurement confidence
- Media Stop Event Occurred
- IP Packet Drop Count
- IP Byte Count
- IP Packet Count
- IP Byte Rate
- IP DSCP
- IP TTL
- IP Protocol
- Media Byte Count
- TCP Connect Round Trip Delay
- TCP Lost Event Count

### Perf-Monitor Metrics: RTP Profile

- Flow Sampling Start Timestamp
- Loss of measurement confidence
- Media Stop Event Occurred
- IP Packet Drop Count
- IP Byte Count

- IP Packet Count
- IP Byte Rate
- Packet Drop Reason
- IP DSCP
- IP TTL
- IP Protocol
- Media Byte Rate Average
- Media Byte Count
- Media Packet Count
- RTP Interarrival Jitter Average
- RTP Packets Lost
- RTP Packets Expected (pkts):
- RTP Packet Lost Event Count:
- RTP Loss Percent

### System Metrics: INTF Profile

- Collection timestamp
- Octet input at Ingress
- Octet output at Egress
- Packets received with errors at Ingress
- Packets with errors at Egress
- Packets discarded at Ingress
- Packets discarded at Egress
- Ingress interface speed
- Egress interface speed

### System Metrics: CPU Profile

- CPU Utilization (1min)
- CPU Utilization (5min)
- Collection timestamp

### System Metrics: MEMORY Profile

- Processor memory utilization %

- Collection timestamp

**App-Health Metrics: MEDIATRACE-HEALTH Profile**

- Requests Received
- Time Last Request Received
- Initiator of Last Request
- Requests Dropped
- Max Concurrent Sessions supported
- Sessions currently active
- Sessions Teared down
- Sessions Timed out
- Hop Info Requests Received
- Performance Monitor Requests Received
- Performance Monitor Requests failed
- Static Policy Requests Received
- Static Policy Requests Failed
- System Data Requests Received
- System Data Requests Failed
- Application Health Requests Received
- Local route change events
- Time of last route change event
- Number of unknown requests received

# Overview of Configuring Cisco Mediatrace

Information can be retrieved from Mediatrace by using in either:

- A pre-scheduled, recurring monitoring session.
- An one-shot, on-demand collection of statistics, known as a Mediatrace poll.

Before you can implement a Mediatrace session or poll, you enable Mediatrace on each network node that you want to collect flow information from. You must enable the Mediatrace Initiator on the network node that you will use to configure, initiate, and control the Mediatrace sessions or polls. On each of the network nodes that you want top collect information from, you must enable the Mediatrace Responder.

To configure a Cisco Mediatrace session, you can set session parameters by associating either of two types of pre-packaged profiles with the session:

- video-monitoring profiles

- system-data profiles

You can also configure your own parameters for a Cisco Mediatrace session by configuring the following types of profiles and associating them with the session:

- Path-specifier profile
- Flow-specifier profile
- Sessions-parameters profile

Therefore, the next section describes how to perform the following tasks in order to configure a Cisco Mediatrace session:

1. Enable mediatrace
2. Setup a video-monitoring profile
3. Setup a system-data profile
4. Setup a path-specifier profile
5. Setup a flow-specifier profile
6. Setup a sessions-params profile
7. Associate profiles with a mediatrace session
8. Schedule a mediatrace session

The next section also describes how to execute a mediatrace poll, which is an on-demand fetch of data from the hops on a specific path.

In addition, the next section describes how to manage mediatrace sessions by performing the following tasks:

- Clear incomplete Cisco Mediatrace sessions
- Troubleshoot a Cisco Mediatrace session

# Limitations

- Mediatrace does not support IPv6.
- Resource Reservation Protocol (RSVP) does not forward an incoming Path message on the same interface (i.e., through the interface from where it receives the path message). It displays an error some message on the console, “ingress interface = egress interface”. But the Path is sent out on the incoming interface in case of an Performance Routing (PfR) border router.

# How to Configure Cisco Mediatrace

## Enabling Cisco Mediatrace

For each node you want to monitor using Cisco Mediatrace, you must enable at least the Cisco Mediatrace Responder. You must also enable the Cisco Mediatrace Initiator for all nodes that you want to initiate Mediatrace sessions or polls.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace initiator** {**source-ip** ip-address | **source-interface** *interface-name*} [**force**] **[max-sessions** *number* **]**
4. **mediatrace responder [max-sessions** *number* **]**
5. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **mediatrace initiator** {**source-ip** ip-address \| **source-interface** *interface-name*} [**force**] **[max-sessions** *number* **]**<br>**Example:**<br>`Router(config)# mediatrace initiator source-ip 10.10.1.1 max-sessions 4` | Enables the Cisco Mediatrace or initiator. You can also use the following keywords:<br>• *ip-address* --Any reachable IP address.<br>• *interface-name* --Any local interface that connects to the initiator.<br>• **max-sessions** --Sets the number of Cisco Mediatrace sessions. |
| **Step 4** | **mediatrace responder [max-sessions** *number* **]**<br>**Example:**<br>`Router(config)# mediatrace responder max-sessions 4` | Enables the Cisco Mediatrace responder. You can also use the following keywords:<br>• **max-sessions** --Sets the number of Cisco Mediatrace sessions. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | **end**<br>**Example:**<br>`Router(config)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace responder app-health**command to verify whether the responder is collecting events, requests, and other Cisco Mediatrace related statistics properly.

For more information about this command, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Configuring a Cisco Mediatrace Video Profile on the Mediatrace Initiator

Cisco Mediatrace provides pre-packaged video-monitoring profiles that contain all of the parameter settings you need to start a video media monitoring session. You can also configure your own video-monitoring profiles on the Mediatrace Initiator.

To initiate a new video media monitoring session, you can associate one of these profiles with a Cisco Mediatrace session when you configure it.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace profile perf-monitor** *name*
4. admin-params
5. **sampling-interval** *seconds*
6. **exit**
7. **metric-list** {**tcp** | **rtp**}
8. **clock-rate** {*type-number* | *type-name*} *rate*
9. **max-dropout** *number*
10. **max-reorder** *number*
11. **min-sequential** *number*
12. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **mediatrace profile perf-monitor** *name*<br>**Example:**<br>`Router(config)# mediatrace profile perf-monitor vprofile-2` | Enters perf-prof configuration mode so that you can configure parameters for a Cisco Mediatrace pre-packaged video-monitoring profile. |
| **Step 4** | admin-params<br>**Example:**<br>`Router(config-mt-prof-perf)# admin-params` | Enters admin parameters configuration mode so that you can configure video-monitoring admin parameters. |
| **Step 5** | **sampling-interval** *seconds*<br>**Example:**<br>`Router(config-mt-prof-perf-params)# sampling-interval 40` | Specifies the interval, in seconds, between samples taken of video-monitoring metrics. |
| **Step 6** | **exit**<br>**Example:**<br>`Router(config-mt-prof-perf-params)# exit` | Exits the current configuration mode and returns to perf-prof configuration mode. |
| **Step 7** | **metric-list** {**tcp** \| **rtp**}<br>**Example:**<br>`Router(config-mt-prof-perf)# metric-list rtp` | Specifies whether the metrics being monitored are for TCP or RTP. |
| **Step 8** | **clock-rate** {*type-number* \| *type-name*} *rate*<br>**Example:**<br>`Router(config-mt-prof-perf-rtp-params)# clock-rate 64` | (Optional) Specifies the clock rate used to sample RTP video-monitoring metrics. Each payload type has a specific clock rate associated with it and is can specified with either a type number or type name. For the available values of the payload type name, see the Cisco Media Monitoring Command Reference . |
| **Step 9** | **max-dropout** *number*<br>**Example:**<br>`Router(config-mt-prof-perf-rtp-params)# max-dropout 2` | (Optional) Specifies the maximum number of dropouts allowed when sampling RTP video-monitoring metrics. Dropouts are the number of packets to ignore ahead the current packet in terms of sequence number. |
| **Step 10** | **max-reorder** *number*<br>**Example:** | (Optional) Specifies the maximum number of reorders allowed when sampling RTP video-monitoring metrics. Reorders are the number of packets to ignore behind the current packet in terms of sequence number. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router(config-mt-prof-perf-rtp-params)# max-reorder 4` | |
| **Step 11** | **min-sequential** *number*<br>**Example:**<br>`Router(config-mt-prof-perf-rtp-params)# min-sequential 2` | (Optional) Specifies the minimum number of packets in a sequence used to classify a RTP flow . |
| **Step 12** | **end**<br>**Example:**<br>`Router(config-mt-prof-perf-rtp-params)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace profile perf-monitor** command to verify that the parameter values for your pre-packaged video-monitoring profiles are set correctly.

For more information about this command, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Configuring a Cisco Mediatrace System Profile

Cisco Mediatrace provides pre-packaged system-data monitoring profiles that contain all of the parameter settings you need to start a system-data monitoring session. You can also configure your own system-data monitoring profiles. To initiate a new system-data monitoring session, you can associate one of these profiles with a Cisco Mediatrace session when you configure it.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace profile system** *name*
4. **metric-list** {**intf** | **cpu** | **memory**}
5. **end**

### DETAILED STEPS

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

| | Command or Action | Purpose |
|---|---|---|
| | `Router# configure terminal` | |
| **Step 3** | **mediatrace profile system** *name*<br>**Example:**<br>`Router(config)# mediatrace profile system system-2` | Enters system profile configuration mode so that you can configure parameters for a Cisco Mediatrace system profile. |
| **Step 4** | **metric-list** {**intf** \| **cpu** \| **memory**}<br>**Example:**<br>`Router(config-sys)# metric-list memory` | Specifies whether the metrics being monitored are for interfaces, the CPU, or the memory. |
| **Step 5** | **end**<br>**Example:**<br>`Router(config-sys)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace profile system**commandto verify that the parameter values for your pre-packaged system-data profiles are set correctly.

For more information about this command, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Configuring a Cisco Mediatrace Path-Specifier Profile

A Cisco Mediatrace session configuration requires a path-specifier profile which defines the parameters that are used to discover the network hops that will be monitored for troubleshooting. The RSVP transport protocol, specified by optional **disc-proto** keyword, is used to do this hop discovery. The parameter values for the flow-specifier should match the values for the media flow that will be traced.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace path-specifier** *name* [**disc-proto rsvp**] {**gsid** *gsid* | **destination ip** *ip-address* **port** *nnnn* }
4. **source ip** *ip-address* **port** *nnnn*
5. **l2-params gateway** *ip-address* **vlan** *vlan-id*
6. **gsid** *gsid*
7. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable** | Enables privileged EXEC mode. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router> enable` | • Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **mediatrace path-specifier** *name* [**disc-proto rsvp**] {**gsid** *gsid* \| **destination ip** *ip-address* **port** *nnnn* }<br>**Example:**<br>`Router(config)# mediatrace path-specifier path-4 disc-proto rsvp destination ip 10.1.1.1 port 400` | Enters path-specifier configuration mode so that you can configure parameters for a Cisco Mediatrace path-specifier profile. This command requires the name, destination address, and port of the path. |
| **Step 4** | **source ip** *ip-address* **port** *nnnn*<br>**Example:**<br>`Router(config-mt-path)# source ip 10.1.1.2 port 600` | Specifies the IP address of the source of the metrics being monitored. |
| **Step 5** | **l2-params gateway** *ip-address* **vlan** *vlan-id*<br>**Example:**<br>`Router(config-mt-path)# l2-params gateway 10.10.10.4 vlan 22` | Specifies the IP address and ID of the virtual LAN of the level-2 gateway.<br>**Note** This command is available only on Catalyst platforms. |
| **Step 6** | **gsid** *gsid*<br>**Example:**<br>`Router(config-mt-path)# gsid 60606060` | Specifies the metadata global session ID of the flow being monitored. |
| **Step 7** | **end**<br>**Example:**<br>`Router(config-mt-path)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace path-specifier** command to verify that the parameter values for your path-specifier profiles are set correctly.

For more information about this command, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Configuring a Cisco Mediatrace Flow-Specifier Profile

A Cisco Mediatrace session configuration requires a flow-specifier profile which defines the source IP address, destination IP address, source port, destination port, and protocol that identifies a flow. You can associate a profile with an actual Cisco Mediatrace session later when you configure it

For RTP media flows, select UDP as protocol.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace flow-specifier** *name*
4. **source-ip** *ip-address* [**source-port** *port* ]
5. **dest-ip** *ip-address* [**dest-port** *port* ]
6. **gsid** *gsid*
7. **ip-protocol** {**tcp** | **udp**}
8. **end**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **mediatrace flow-specifier** *name*<br>**Example:**<br>`Router(config)# mediatrace flow-specifier flow-6` | Enters flow-specifier configuration mode so that you can configure parameters for a Cisco Mediatrace flow-specifier profile. |
| **Step 4** | **source-ip** *ip-address* [**source-port** *port* ]<br>**Example:**<br>`Router(config-mt-flowspec)# source-ip 10.1.1.2 source-port 600` | (Optional) Specifies the IP address of the source of the metrics being monitored. |
| **Step 5** | **dest-ip** *ip-address* [**dest-port** *port* ]<br>**Example:**<br>`Router(config-mt-flowspec)# dest-ip 10.1.1.2 dest-port 600` | Specifies the IP address of the destination of the metrics being monitored. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | **gsid** *gsid*<br>**Example:**<br>`Router(config-mt-flowspec)# gsid 60606060` | Specifies the metadata global session ID of the flow being monitored. |
| **Step 7** | **ip-protocol** {**tcp** \| **udp**}<br>**Example:**<br>`Router(config-mt-flowspec)# ip-protocol tcp` | Specifies whether the metrics being monitored are for TCP or UDP. |
| **Step 8** | **end**<br>**Example:**<br>`Router(config-mt-flowspec)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace flow-specifier** command to verify that the parameter values for your flow-specifier profiles are set correctly.

For more information about this command, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Configuring a Cisco Mediatrace Session Parameters Profile

A Cisco Mediatrace session configuration requires a session-params profile, which defines the characteristics of a Cisco Mediatrace session and help it to operate smoothly. You can associate a profile with an actual Cisco Mediatrace session later when you configure it

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace session-params** *name*
4. **response-timeout** *seconds*
5. **frequency** {*frequency* | **on-demand**} **inactivity-timeout** *seconds*
6. **history** *buckets*
7. **route-change reaction-time** *seconds*
8. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:** | Enables privileged EXEC mode.<br>• Enter your password if prompted. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router> enable` | |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **mediatrace session-params** *name*<br>**Example:**<br>`Router(config-mt-sesparam)# mediatrace session-params qos-2` | Enters session-params configuration mode so that you can configure parameters for a Cisco Mediatrace session-params profile. |
| **Step 4** | **response-timeout** *seconds*<br>**Example:**<br>`Router(config-mt-sesparam)# response-timeout 8` | Specifies the amount of time, in seconds, the initiator will wait for a response from the responder. |
| **Step 5** | **frequency** {*frequency* \| **on-demand**} **inactivity-timeout** *seconds*<br>**Example:**<br>`Router(config-mt-sesparam)# frequency 4 inactivity-timeout 2` | Specifies the interval, in seconds, between samples taken of session-params metrics and the amount of time, in seconds, the initiator will remain active without any activity from the responder. |
| **Step 6** | **history** *buckets*<br>**Example:**<br>`Router(config-mt-sesparam)# history 2` | Specifies the number of historical data sets kept, up to a maximum of ten. |
| **Step 7** | **route-change reaction-time** *seconds*<br>**Example:**<br>`Router(config-mt-sesparam)# route-change reaction-time 8` | Specifies the amount of time, in seconds, the initiator will wait for the responder to react to its additional route changes. The range is seconds. |
| **Step 8** | **end**<br>**Example:**<br>`Router(config-mt-sesparam)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace session-param**command to verify that the parameter values for your session-parameters profiles are set correctly.

For more information about this command, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Configuring a Cisco Mediatrace Session

The Cisco Mediatrace session configuration links the various profiles to a session. Only one of each type of profile can be associated with a Cisco Mediatrace session.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace** *session-number*
4. **trace-route**
5. **path-specifier** {[ **forward** ] *path-name* | **reverse** *path-name* }
6. **session-params** *name*
7. **profile system** *name*
8. **profile perf-monitor** *name* **flow-specifier** *flow-specifier-name*
9. **profile snmp** *name*
10. **profile custom** *name*
11. **last-node** { **auto** | **address** *address* }
12. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **mediatrace** *session-number*<br>**Example:**<br>`Router(config)# mediatrace 157` | Enters session configuration mode. |
| **Step 4** | **trace-route**<br>**Example:**<br>`Router(config-mt-session)# trace-route` | Enables the running of trace route for the Cisco Mediatrace session. By default trace route is enabled. To stop running trace route, use the **no** form of this command. |
| **Step 5** | **path-specifier** {[ **forward** ] *path-name* \| **reverse** *path-name* }<br>**Example:**<br>`Router(config-mt-session)# path-specifier path-4` | Associates a path-specifier profile with the Cisco Mediatrace session. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | **session-params** *name*<br>**Example:**<br>`Router(config-mt-session)# session-params session-6` | Associates a session-parameters profile with the Cisco Mediatrace session. |
| **Step 7** | **profile system** *name*<br>**Example:**<br>`Router(config-mt-session)# profile system sys-2` | Associates a system profile with the Cisco Mediatrace session. |
| **Step 8** | **profile perf-monitor** *name* **flow-specifier** *flow-specifier-name*<br>**Example:**<br>`Router(config-mt-session)# profile perf-monitor monitor-6 flow-specifier flow-4` | Associates a perf-monitor profile and flow-specifier with the Cisco Mediatrace session. |
| **Step 9** | **profile snmp** *name*<br>**Example:**<br>`Router(config-mt-session)# profile snmp snmp-2` | Associates an SNMP profile with the Cisco Mediatrace session. |
| **Step 10** | **profile custom** *name*<br>**Example:**<br>`Router(config-mt-session)# profile custom cp-2` | Associates an SNMP profile with the Cisco Mediatrace session. |
| **Step 11** | **last-node** { **auto** \| **address** *address* }<br>**Example:**<br>`Router(config-mt-session)# last-node address 10.1.1.1` | Configures the last node for the Cisco Mediatrace session. |
| **Step 12** | **end**<br>**Example:**<br>`Router(config-mt-session)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace session**command to display the parameter settings for a specific session or all sessions.

Use the **show mediatrace responder app-health** command and the **show mediatrace responder sessions**command to determine the status of the nodes being monitored.

If Cisco Mediatrace is not collecting all of the data that you want, use the **debug mediatrace**command.

For more information about these commands, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Scheduling a Cisco Mediatrace Session

Once you have configured a Cisco Mediatrace session, you can schedule it to begin when you want to start collecting the data. If the Cisco Mediatrace session is designed to collect performance monitoring metrics, it goes out to enable the Performance Monitor when the session begins.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mediatrace schedule** *session ID* [*life* {**forever** | *secs*}] [**start-time** {*hh*:*mm*:[:*ss*][*month day*| *day month*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *secs*] [**recurring**]
4. **end**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **mediatrace schedule** *session ID* [*life* {**forever** | *secs*}] [**start-time** {*hh*:*mm*:[:*ss*][*month day*| *day month*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *secs*] [**recurring**]<br>**Example:**<br>`Router(config)# mediatrace schedule 22 life 40 start-time 10:00:00 AUG 20 recurring` | Specifcies when the session will occur. Use these settings:<br>• *session ID* --Which session to run.<br>• **life** --Amount of time the session lasts, either the number of seconds or forever.<br>• **start-time** --When the session starts, whether it is at a specified time and date, pending an event, immediately, or after a specified time and date.<br>• **ageout** --Timeout before removing the session configuration on the initiator.<br>• **recurring** --Session reoccurs at the specified time. |
| **Step 4** | **end**<br>**Example:**<br>`Router(config-mt-sched)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

Use the **show mediatrace session**command to verify that the intended values are set for the parameters for a specific session or all sessions.

Use the **show mediatrace responder app-health** command and the **show mediatrace responder sessions**command to determine the status of the nodes being monitored.

If Cisco Mediatrace is not collecting all of the data that you want, use the **debug mediatrace**command.

For more information about these commands, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Clearing a Cisco Mediatrace Session

You can clear incomplete mediatrace sessions on the Initiator by using the **clear mediatrace incomplete-sessions**command as described below. This coammand also cleans up all Performance Monitor settings that were configured by Cisco Mediatrace. For sessions created by the config commands, use the **no mediatrace schedule**command. The cleanup triggers a "session teardown" message to RSVP followed by a cleanup of the local mediatrace sessions database.

### SUMMARY STEPS

1. **enable**
2. **clear mediatrace incomplete-sessions**
3. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **clear mediatrace incomplete-sessions**<br>**Example:**<br>`Router# clear mediatrace incomplete-sessions` | Clears incomplete mediatrace sessions. |
| **Step 3** | **end**<br>**Example:**<br>`Router# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the status of your Cisco Mediatrace session, use the **show mediatrace responder sessions**command.

For more information about these commands, see the How to Troubleshoot and Monitor a Cisco Mediatrace Session, on page 25.

# Executing a Cisco Mediatrace Poll

Cisco Mediatrace polls are used to perform an on-demand fetch of data from the hops on a specific path. Some examples of how it can be used are:

- To retrieve data using a pre-configured session. In this case, no other parameters have to be specified inline. The pre-configured session must be have the frequency type set to on-demand.
- To retrieve the system data, hop or video monitoring information from hops along the specified path. You can specify the path as a pre-configured path-specifier or an inline path specification, in case you do not have config mode privileges. Note that by default, Cisco Mediatrace tries to configure nodes along the path to report passive monitoring metrics, and then waits for a configurable amount of time before going out again to collect the data.
- The **configless** keyword can be used to fetch data from the nodes along a media path, which already have Performance Monitor policies configured using the Performance Monitor commands. Some key things to keep in mind when fetching data using this method are that:
  - The default perf-monitor profile or associated perf-monitor profile will have a sampling interval. If the sampling interval of the static policy does not match the one in the associated perf-monitor profile, no data is returned.
  - If there is no Performance Monitor policy configured on a Responder node, the Cisco Mediatrace responder does not try to configure Performance Monitor and simply reports error to the initiator.

### SUMMARY STEPS

1. **enable**
2. **mediatrace poll** {**no-traceroute** | **session** *number* | [**timeout** *value*] **path-specifier**{**name** *path-name* | **gsid** *gsid* | {[**disc-proto rsvp**] **destination ip** *ip-address* [**port** *nnnnn*] | **source ip** *ip-address* [**port** *nnnnn*] **destination ip** *ip-address* [**port** *nnnn*] [**ip-protocol**{**tcp** | **udp**}]} {**app-health** | **hops** | **l2-params gateway** *ip-address* | **system** [**profile** *system-profile-name*] | [**configless**] **perf-monitor** [**profile** *profile-name*]} {**flow-specifier** *name* | **source-ip** *ipaddress* [**source-port** *nnnnn*] **dest-ip** *ipaddress* [**dest-port** *nnnnn*] **ip-protocol** {**tcp** | **udp**}}}}
3. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **mediatrace poll** {**no-traceroute** \| **session** *number* \| [**timeout** *value*] **path-specifier**{**name** *path-name* \| **gsid** *gsid* \| {[**disc-proto rsvp**] **destination ip** *ip-address* [**port** *nnnnn*] \| **source ip** *ip-address* [**port** *nnnnn*] **destination ip** *ip-address* [**port** *nnnn*] [**ip-protocol**{**tcp** \| **udp**}]} {**app-health** \| **hops** \| **l2-params gateway** *ip-address* \| **system** [**profile** *system-profile-name*] \| [**configless**] | Performs an on-demand fetch of data from the hops on a specific path. You can specify the hops using one of the following types of information:<br>• A session definition or its constituent parameters<br>• A system profile definition or its constituent parameters |

| | Command or Action | Purpose |
|---|---|---|
| | **perf-monitor** [**profile** *profile-name*]} {**flow-specifier** *name* \| **source-ip** *ipaddress* [**source-port** *nnnnn*] **dest-ip** *ipaddress* [**dest-port** *nnnnn*] **ip-protocol** {**tcp** \| **udp**}}}}<br>**Example:**<br>**Example:**<br>`Router# mediatrace poll session 22` | • A combination of a path-specifier profile definition and a perf-monitor profile definition or their constituent parameters<br>**Note** The **l2-params gateway** keyword is available only on Catalyst platforms. |
| **Step 3** | **end**<br>**Example:**<br>`Router# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

If Cisco Mediatrace is not collecting all of the data that you want:

- Use the **show mediatrace session**command to verify that the intended values are set for the parameters for a specific session or all sessions.
- Use the **show mediatrace responder app-health** command and the **show mediatrace responder sessions**command to determine the status of the nodes being monitored.
- Use the **debug mediatrace**command to view error messages.

## Examples

**Tip** For examples of poll output, see Configuration Examples for Cisco Mediatrace, on page 32.

The following example shows how to fetch the default system metrics when the source IP address, source port, and destination port are not known. Cisco Mediatrace uses the best local IP address as source IP address to find which hops are using RSVP.

**mediatrace poll path dest** *ip-address* **system**

The following example shows how to fetch the default system metrics when the source and destination port numbers are not known. RSVP finds the hop between the specified source and destination.

**mediatrace poll path source** *ip-address* **dest** *ip-address* **system**

The following example shows how to fetch the default system metrics when the source and destination port numbers are known. RSVP finds the hop using this information.

**mediatrace poll path source-ip** *ip-address* **source - port** *nnnn* **dest-ip** *ip-address* **dest - port** *nnnn* **ip-protocol udp system**

The following example shows how to fetch the default set of RTP metrics when the source and destination port numbers are not known. Cisco Mediatrace uses the path source and destination IP addresses to find the hops as well as filter the Performance Monitor data.

**mediatrace poll path source** *ip-address* **dest** *ip-address* **perf-monitor**

The following example shows how to fetch the default set of RTP metrics. Cisco Mediatrace uses the path parameters to discover hops and uses the inline flow specifier profile as a filter for Performance Monitor data.

**mediatrace poll path source** *ip-address* **dest** *ip-address* **perf-monitor source-ip** *ip-address* **source - port** *nnnn* **dest-ip** *ip-address* **dest - port** *nnnn* **ip-protocol udp**

The following example shows how to fetch the default set of TCP metrics. Cisco Mediatrace uses the path parameters to discover hops and uses the inline flow-specifier profile as a filter for Performance Monitor data.

**mediatrace poll path source** *ip-address* **dest** *ip-address* **perf-monitor source-ip** *ip-address* **source - port** *nnnn* **dest-ip** *ip-address* **dest - port** *nnnn* **ip-protocol tcp**

The following example shows how to fetch the default set of RTP metrics. Cisco Mediatrace uses the best local IP address as source IP address for finding hops on the path and uses the inline flow specifier profile as a filter for Performance Monitor data.

**mediatrace poll path dest** *ip-address* **perf-monitor source-ip** *ip-address* **source - port** *nnnn* **dest-ip** *ip-address* **dest - port** *nnnn* **ip-protocol udp**

The following example shows how to fetch the default set of TCP metrics. Cisco Mediatrace uses the best local IP address as source IP address for finding hops on the path and uses the inline flow-specifier profile as a filter for Performance Monitor data.

**mediatrace poll path dest** *ip-address* **perf-monitor source-ip** *ip-address* **source - port** *nnnn* **dest-ip** *ip-address* **dest - port** *nnnn* **ip-protocol tcp**

The following example shows how to fetch the default set of RTP metrics from the static policy that is already configured on the hops. The command does not configure the Performance Monitor. Cisco Mediatrace uses the path parameters to discover hops and use the inline flow specifier profile as a filter for Performance Monitor data.

**mediatrace poll path source** *ip-address* **dest** *ip-address* **configless perf-monitor flow-specifier source** *ip-address* **port** *nnnn* **dest** *ip-address* **port** *nnnn* **ip-protocol udp**

### Poll Output Example

This example shows the output is produced by the following hops poll command:

```
mediatrace poll path-specifier source 10.10.130.2 destination 10.10.132.2 hops
Started the data fetch operation.
Waiting for data from hops.
This may take several seconds to complete...
Data received for hop 1
Data received for hop 2
Data fetch complete.
Results:
Data Collection Summary:
  Request Timestamp: 22:47:56.788 PST Fri Oct 29 2010
  Request Status: Completed
  Number of hops responded (includes success/error/no-record): 2
  Number of hops with valid data report: 2
  Number of hops with error report: 0
  Number of hops with no data record: 0
Detailed Report of collected data:
    Number of Mediatrace hops in the path: 2
    Mediatrace Hop Number: 1 (host=responder1, ttl=254)
      Reachability Address: 10.10.12.3
      Ingress Interface: Gi0/1
      Egress Interface: Gi0/2
    Mediatrace Hop Number: 2 (host=responder2, ttl=253)
```

```
Reachability Address: 10.10.34.3
Ingress Interface: Gi0/1
Egress Interface: Gi0/2
```

# How to Troubleshoot and Monitor a Cisco Mediatrace Session

Use the **show** commands described in this section to troubleshoot to monitor a Cisco Mediatrace session.

**Tip** For sample outputs, see the Examples section, in this chapter.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **show mediatrace profile perf-monitor** [*name*]
4. **show mediatrace profile system** [*name*]
5. **show mediatrace flow-specifier** [*name*]
6. **show mediatrace path-specifier** [*name*]
7. **show mediatrace initiator**
8. **show mediatrace session-params** [*name*]
9. **show mediatrace session** [**config**| **data**| **stats**| **hops**] [**brief**| *ID*]
10. **show mediatrace responder app-health**
11. **show mediatrace responder sessions [** *global-session-id* **| brief | details]**
12. **debug mediatrace** {**event** | **trace** | **error**} [**initiator** | **responder**| *session-id*]
13. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **show mediatrace profile perf-monitor** [*name*]<br>**Example:**<br>`Router(config)# show mediatrace profile perf-monitor vprofile-4` | Displays the parameters configured for all pre-packaged video-monitoring profiles or the specified profile. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | **show mediatrace profile system** [*name*]<br>**Example:**<br>`Router(config)# show mediatrace profile system system-8` | Displays the parameters configured for all pre-packaged system-data monitoring profiles or the specified profile. |
| **Step 5** | **show mediatrace flow-specifier** [*name*]<br>**Example:**<br>`Router(config)# show mediatrace flow-specifier flow-2` | Displays the parameters configured for all flow-specifier profiles or the specified flow-specifier profile. |
| **Step 6** | **show mediatrace path-specifier** [*name*]<br>**Example:**<br>`Router(config)# show mediatrace path-specifier path-6` | Displays the parameters configured for all path-specifier profiles or the specified path-specifier profile. |
| **Step 7** | **show mediatrace initiator**<br>**Example:**<br>`Router(config)# show mediatrace initiator` | Displays the parameters configured for the initiator profile. |
| **Step 8** | **show mediatrace session-params** [*name*]<br>**Example:**<br>`Router(config)# show mediatrace session-params sysparams-2` | Displays the monitoring parameters for the session like frequency, response timeout, ands so on.<br>the parameters configured for all pre-packaged system-data monitoring profiles or the specified profile. |
| **Step 9** | **show mediatrace session** [**config**\| **data**\| **stats**\| **hops**] [**brief**\| *ID*]<br>**Example:**<br>`Router(config)# show mediatrace session data 1002` | Displays the parameters configured for all session profiles or the specified session profile. Use the following keywords to display the corresponding information:<br>• **config** --Configuration of the session.<br>• **data** --All data records collected and still cached at the Initiator.<br>• **stats** --Statistics for this service path or session.<br>• **hops** --Prior service paths (if available) and current service paths discovered. Also shows where and when the last route change happened.<br>• **brief** -- Only a list of sessions with ID, destination/source address/port, and their role association as Initiator or Responder.<br>• *ID* -- Session ID and some state information. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 10** | **show mediatrace responder app-health**<br>**Example:**<br>`Router(config)# show mediatrace responder app-health` | Displays the current status of the responder. |
| **Step 11** | **show mediatrace responder sessions [** *global-session-id* **\| brief \| details]**<br>**Example:**<br>`Router(config)# show mediatrace responder sessions` | Displays the information about all or specific active sessions on local responder. Use the following keywords to display the corresponding information<br>• *global-session-id* -- ID of the session for which information is displayed.<br>• **brief** --Displays only the destination and source address/port of the path, their role as either Initiator or Responder, and some state information.<br>• **details** --Displays all information. |
| **Step 12** | **debug mediatrace** {**event** \| **trace** \| **error**} [**initiator** \| **responder**\| *session-id*]<br>**Example:**<br>`Router(config)# debug mediatrace event 24` | Enables debugging for a particular path, or a particular session, or for all Initiator and Responder functions. You can use the following options:<br>• **event** -- Displays only event information.<br>• **trace** -- Displays only trace information.<br>• **error** -- Displays only errors.<br>• **initiator** -- Displays information for only the initiator.<br>• **responder** -- Displays information for only the responder.<br>• *session-id* -- Displays information for only the session. |
| **Step 13** | **end**<br>**Example:**<br>`Router(config)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

### Examples

**Note** For a complete description of the output for the following show commands, see the *Cisco Media Monitoring Command Reference*.

The following example displays video-monitoring profiles:

```
Router# show mediatrace profile perf-monitor
Perf-monitor Profile: vprof-4
Metric List: rtp
RTP Admin Parameter:
  Max Dropout: 5
  Max Reorder: 5
  Min Sequential: 5
Admin Parameter:
  Sampling Interval (sec): 30
```

The following example displays system-data profiles:

```
Router# show mediatrace profile
system

System Profile: sys-1
Metric List: intf
```

The following example displays flow-specifier profiles:

```
Router# show mediatrace
 flow-specifier flow-1
Flow Specifier: flow-1
    Source address/port:
    Destination address/port:
    Protocol: udp
```

The following example displays path-specifier profiles:

```
Router# show mediatrace
 path-specifier flow-1
Path Configuration: ps1
    Destination address/port: 10.10.10.1
    Source address/port: 10.10.10.4
    Gateway address/vlan:
    Discovery protocol: rsvp
```

The following example displays the initiator profile:

```
Router# show mediatrace
 initiator
Version: Mediatrace 1.0
Mediatrace Initiator status: enabled
Source IP: 1.1.1.1
Number of Maximum Allowed Active Session: 127
Number of Configured Session: 1
Number of Active Session    : 0
Number of Pending Session   : 0
Number of Inactive Session  : 1
Note: the number of active session may be higher than max active session
      because the max active session count was changed recently.
```

The following example displays session profiles:

```
Router# show mediatrace session-params
Session Parameters: s-1
    Response timeout (sec): 60
    Frequency: On Demand
```

```
    Inactivity timeout (sec): 300
    History statistics:
       Number of history buckets kept: 3
    Route change:
       Reaction time (sec): 5
```

The following example displays Mediatrace session statistics:

```
Router# show mediatrace session stats 2
Session Index: 2
Global Session Id: 86197709
Session Operation State: Active
Operation time to live: Forever
Data Collection Summary:
  Request Timestamp: 23:55:04.228 PST Fri Oct 29 2010
  Request Status: Completed
  Number of hops responded (includes success/error/no-record): 2
  Number of Non Mediatrace hops responded: 0
  Number of hops with valid data report: 2
  Number of hops with error report: 0
  Number of hops with no data record: 0
Detailed Report of collected data:
  Last Route Change Timestamp:
  Route Index: 0
    Number of Mediatrace hops in the path: 2
    Mediatrace Hop Number: 1 (host=responder1, ttl=254)
      Metrics Collection Status: Success
      Reachability Address: 10.10.12.3
      Ingress Interface: Gi0/1
      Egress Interface: Gi0/2
      Traceroute data:
        Address List: 1.2.2.3
        Round Trip Time List (msec): 12 msec
```

**Note** The rest of the data for hop 1 is similar to the data for hop 2, as shown below.

```
Mediatrace Hop Number: 2 (host=responder2, ttl=253)
      Metrics Collection Status: Success
      Reachability Address: 10.10.34.3
      Ingress Interface: Gi0/1
      Egress Interface: Gi0/2
      Metrics Collected:
        Collection timestamp: 23:55:04.237 PST Fri Oct 29 2010
        Octet input at Ingress (KB): 929381.572
        Octet output at Egress (MB): 1541.008502
        Pkts rcvd with err at Ingress (pkts): 0
        Pkts errored at Egress (pkts): 0
        Pkts discarded at Ingress (pkts): 0
        Pkts discarded at Egress (pkts): 0
        Ingress i/f speed (mbps): 1000.000000
        Egress i/f speed (mbps): 1000.000000
```

The following example displays Mediatrace session configuration information:

```
Router# show mediatrace session config 2
Global Session Id: 93642270
---------------------------
Session Details:
    Path-Specifier: ps1
```

```
    Session Params: sp1
    Collectable Metrics Profile: intf1
    Flow Specifier:
Schedule:
   Operation frequency (seconds): 30  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
History Statistics:
   Number of history Buckets kept: 10
```

The following example displays Mediatrace session hops:

```
show mediatrace session hops 2
Session Index: 2
Global Session Id: 93642270
Session Operation State: Active
Data Collection Summary:
  Request Timestamp: 13:40:32.515 PST Fri Jun 18 2010
  Request Status: Completed
  Number of hops responded (includes success/error/no-record): 3
  Number of hops with valid data report: 3
  Number of hops with error report: 0
  Number of hops with no data record: 0
Detailed Report of collected data:
  Last Route Change Timestamp:
  Route Index: 0
    Number of Mediatrace hops in the path: 3
    Mediatrace Hop Number: 1 (host=responder1, ttl=254)
      Ingress Interface: Gi0/1
      Egress Interface: Gi1/0
    Mediatrace Hop Number: 2 (host=responder2, ttl=253)
      Ingress Interface: Gi0/1
      Egress Interface: Gi1/0
    Mediatrace Hop Number: 3 (host=responder3, ttl=252)
      Ingress Interface: Gi0/1
      Egress Interface: Gi0/2
```

The following example displays Mediatrace session data:

```
Router# show mediatrace session data 2
Session Index: 2
Global Session Id: 35325453
Session Operation State: Active
Bucket index: 1
Data Collection Summary:
  Request Timestamp: 13:02:47.969 PST Fri Jun 18 2010
  Request Status: Completed
  Number of hops responded (includes success/error/no-record): 3
  Number of hops with valid data report: 3
  Number of hops with error report: 0
  Number of hops with no data record: 0
Detailed Report of collected data:
  Last Route Change Timestamp:
  Route Index: 0
    Number of Mediatrace hops in the path: 3
    Mediatrace Hop Number: 1 (host=responder1, ttl=254)
      Metrics Collection Status: Success
      Ingress Interface: Gi0/1
```

```
      Egress Interface: Gi1/0
      Metrics Collected:
        Collection timestamp: 13:04:57.781 PST Fri Jun 18 2010
        Octet input at Ingress (KB): 10982.720
        Octet output at Egress (KB): 11189.176
    Pkts rcvd with err at Ingress (pkts): 0
        Pkts errored at Egress (pkts): 0
        Pkts discarded at Ingress (pkts): 0
        Pkts discarded at Egress (pkts): 0
        Ingress i/f speed (mbps): 1000.000000
        Egress i/f speed (mbps): 1000.000000
    Mediatrace Hop Number: 2 (host=responder2, ttl=253)
      Metrics Collection Status: Success
      Ingress Interface: Gi0/1
      Egress Interface: Gi1/0
      Metrics Collected:
        Collection timestamp: 13:04:57.792 PST Fri Jun 18 2010
        Octet input at Ingress (MB): 1805.552836
        Octet output at Egress (MB): 1788.468650
        Pkts rcvd with err at Ingress (pkts): 0
        Pkts errored at Egress (pkts): 0
        Pkts discarded at Ingress (pkts): 0
        Pkts discarded at Egress (pkts): 0
        Ingress i/f speed (mbps): 1000.000000
        Egress i/f speed (mbps): 1000.000000
```

The following example displays application health information for the Mediatrace responder:

```
Router# show mediatrace responder app-health
Mediatrace App-Health Stats:
   Number of all requests received: 0
   Time of the last request received:
   Initiator ID of the last request received:  0
   Requests dropped due to queue full: 0
   Responder current max sessions: 45
   Responder current active sessions: 0
   Session down or tear down requests received: 0
   Session timed out and removed: 0
   HOPS requests received: 0
   VM dynamic polling requests received: 0
   VM dynamic polling failed: 0
   VM configless polling requests received: 0
   VM configless polling failed: 0
   SYSTEM data polling requests received: 0
   SYSTEM data polling requests failed: 0
   APP-HEALTH polling requests received: 0
   Route Change or Interface Change notices received: 0
   Last time Route Change or Interface Change:
   Unknown requests received: 0
```

The following example displays brief session information for the Mediatrace responder:

```
Router# show mediatrace responder sessions brief
Local Responder configured session list:
Current configured max sessions: 45
Current number of active sessions: 0
session-id initiator-name       src-ip          src-port   dst-ip          dst-port det-l
  2   host-18     10.10.10.2     200   10.10.10.8     200
```

# Configuration Examples for Cisco Mediatrace

## Example Basic Mediatrace Configuration

The topology for this example includes:

- One mediatrace initiator (10.10.12.2)
- Two mediatrace responders between:
  - A media source (10.10.130.2)
  - A destination (10.10.132.2)

In this example, there is an RTP traffic stream from the source (address=10.10.130.2, port=1000, to the destination (address=10.10.132.2, port=2000).

The basic configuration of the mediatrace responder is as follows:

```
mediatrace responder
snmp-server community public RO
```

The basic configuration of the mediatrace initiator is as follows:

```
mediatrace initiator source-ip 10.10.12.2
mediatrace profile system intf1
mediatrace profile perf-monitor rtp1
mediatrace path-specifier path1 destination ip 10.10.132.2 port 2000
 source ip 10.10.130.2 port 1000
mediatrace flow-specifier flow1
 source-ip 10.10.130.2 source-port 1000
 dest-ip 10.10.132.2 dest-port 2000
mediatrace session-params sp1
 response-timeout 10
 frequency 60 inactivity-timeout 180
mediatrace 1
 path-specifier path1
 session-params sp1
 profile perf-monitor rtp1 flow-specifier flow1
mediatrace schedule 1 life forever start-time now
mediatrace 2
 path-specifier path1
 session-params sp1
 profile system intf1
mediatrace schedule 2 life forever start-time now
```

A sample reverse mediatrace configuration is given below.

```
Device# show mediatrace initiator
Mediatrace Initiator Software Version: 3.0
Mediatrace Protocol Version: 1
Mediatrace Initiator status: enabled

Source IP: 10.10.1.1
Source IPv6:

Number of Maximum Allowed Active Session: 8
Number of Configured Session: 3
```

```
Number of Active Session    : 2
Number of Pending Session   : 0
Number of Inactive Session  : 1
Number of Total Proxy Session     : 1
Number of Active Proxy Session    : 1
Number of Pending Proxy Session   : 0
Number of Inactive Proxy Session  : 0

Note: the number of active session may be higher than max active session
      because the max active session count was changed recently.

Device# show run
Device# show running-config | show mediatrace
mediatrace responder
mediatrace initiator source-ip 10.10.1.1
mediatrace profile perf-monitor MT_PERF_RTP
mediatrace path-specifier MT_PATH destination ip 10.11.1.10 port 21064
 source ip 10.10.1.11 port 28938
mediatrace path-specifier MT_PATH2 destination ip 10.10.10.10 port 16514
 source ip 10.10.1.10 port 16558
mediatrace flow-specifier MT_FLOW
 source-ip 10.10.1.11 source-port 28938
 dest-ip 10.10.1.50 dest-port 21064
mediatrace flow-specifier MT_FLOW2
 source-ip 10.1.1.50 source-port 21064
 dest-ip 10.1.1.11 dest-port 28938
mediatrace session-params MT_PARAMS
 response-timeout 50
 frequency 60 inactivity-timeout 180
 history data-sets-kept 10
mediatrace reverse 155
 path-specifier forward/reverse MT_PATH/MT_PATH2
 session-params MT_PARAMS
 profile perf-monitor MT_PERF_RTP flow-specifier MT_FLOW2
mediatrace schedule 155 life forever start-time now
mediatrace 157
 path-specifier MT_PATH
 session-params MT_PARAMS
 profile perf-monitor MT_PERF_RTP flow-specifier MT_FLOW
mediatrace schedule 157 life forever start-time now
```

# Where to Go Next

For more information about configuring the products in the Medianet product family, see the other chapter in this guide or see the *Cisco Media Monitoring Configuration Guide*.

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| Design, configuration, and troubleshooting resources for Cisco Mediatrace and other Cisco Medianet products, including a Quick Start Guide and Deployment Guide. | See the Cisco Medianet Knowledge Base Portal, located at http://www.cisco.com/web/solutions/medianet/knowledgebase/index.html. |
| IP addressing commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples | *Cisco Media Montoring Command Reference* |

### Standards

| Standard | Title |
|---|---|
| No new or modified standards are supported, and support for existing standards has not been modified | -- |

### MIBs

| MIB | MIBs Link |
|---|---|
| No new or modified MIBs are supported, and support for existing MIBs has not been modified | -- |

### RFCs

| RFC[1] | Title |
|---|---|
| RFC 2205 | *RSVP: Resource ReSerVation Protocol*<br>http://www.ietf.org/rfc/rfc2205.txt |

[1] These references are only a sample of the many RFCs available on subjects related to IP addressing and IP routing. Refer to the IETF RFC site at http://www.ietf.org/rfc.html for a full list of RFCs.

### 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.<br>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.<br>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password. | http://www.cisco.com/techsupport |

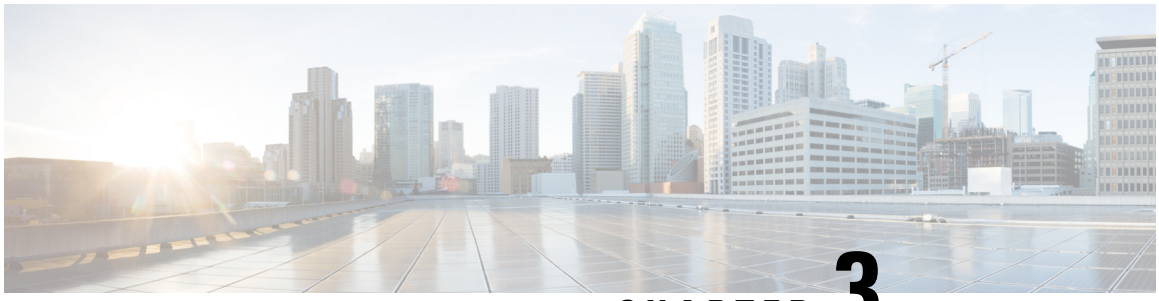
# Feature Information for Cisco Mediatrace

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 1: Feature Information for Cisco Mediatrace*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Cisco Mediatrace 1.0 | 15.1(3)T<br>12.2(58)SE<br>15.1(4)M1<br>15.0(1)SY<br>15.1(1)SY<br>15.1(1)SY1<br>15.2(1)S<br>Cisco IOS XE Release 3.5S<br>15.1(2)SY | This feature enables you to isolate and troubleshoot network degradation problems for data streams.<br>The following commands were introduced or modified by this feature: **admin-params**, **clear mediatrace**, **incomplete-sessions**, **clock-rate** (RTP parameters), **dest-ip** (flow), **frequency** (session parameters), **history** (session parameters), **ip-protocol** (flow), **max-dropout**, **max-reorder**, **mediatrace**, **mediatrace initiator**, **mediatrace responder**, **mediatrace path-specifier**, **mediatrace poll**, **mediatrace profile perf-monitor**, **mediatrace profile system**, **mediatrace schedule**, **mediatrace session-params**, **metric-list** (monitoring profile), **metric-list** (system profile), **min-sequential**, **path-specifier**, **profile perf-monitor**, **profile system**, **response-timeout** (session parameters), **route-change reaction-time**, **sampling-interval**, **session-params**, **show mediatrace flow-specifier**, **show mediatrace initiator**, **show mediatrace path-specifier**, **show mediatrace profile system**, **show mediatrace profile perf-monitor**, **show mediatrace responder app-health**, **show mediatrace responder sessions**, **show mediatrace session**, **show mediatrace session-params**, **source-ip** (flow), and **source ip** (path). |



CHAPTER 3

# Configuring Cisco Performance Monitor

This document contains information about and instructions for configuring Cisco Performance Monitor.



# 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 Cisco Performance Monitor

## Overview of Cisco Performance Monitor

Cisco Performance Monitor enables you to monitor the flow of packets in your network and become aware of any issues that might impact the flow before it starts to significantly impact the performance of the application in question. Performance monitoring is especially important for video traffic because high quality interactive video traffic is highly sensitive to network issues. Even minor issues that may not affect other applications can have dramatic effects on video quality.

Because Cisco Performance Monitor uses similar software components and commands as Cisco NetFlow and Cisco Flexible NetFlow, familiarity with these products will help you to understand how to configure Cisco Performance Monitor. These products provide statistics on packets flowing through a router and are the

standard for acquiring IP operational data from IP networks. They provide data to support network and security monitoring, network planning, traffic analysis, and IP accounting. For more information about Cisco NetFlow and Cisco Flexible NetFlow, see the documents listed in the Additional References section.

For more information about the design, configuration, and troubleshooting of Performance Monitor and other Cisco Medianet products, including a Quick Start Guide and Deployment Guide, see the Cisco Medianet Knowledge Base Portal, located at http://www.cisco.com/web/solutions/medianet/knowledgebase/index.html.

# Prerequisites for Configuring Cisco Performance Monitor

The following prerequisites must be met before you can configure Cisco Performance Monitor:

### 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 Cisco Performance Monitor: Cisco Express Forwarding or distributed Cisco Express Forwarding.

### IPv6 Traffic

- Cisco Express Forwarding must be enabled on your router and on any interfaces on which you want to enable Cisco Performance Monitor. You can use the **ipv6 cef** command to enable Cisco Express Forwarding.

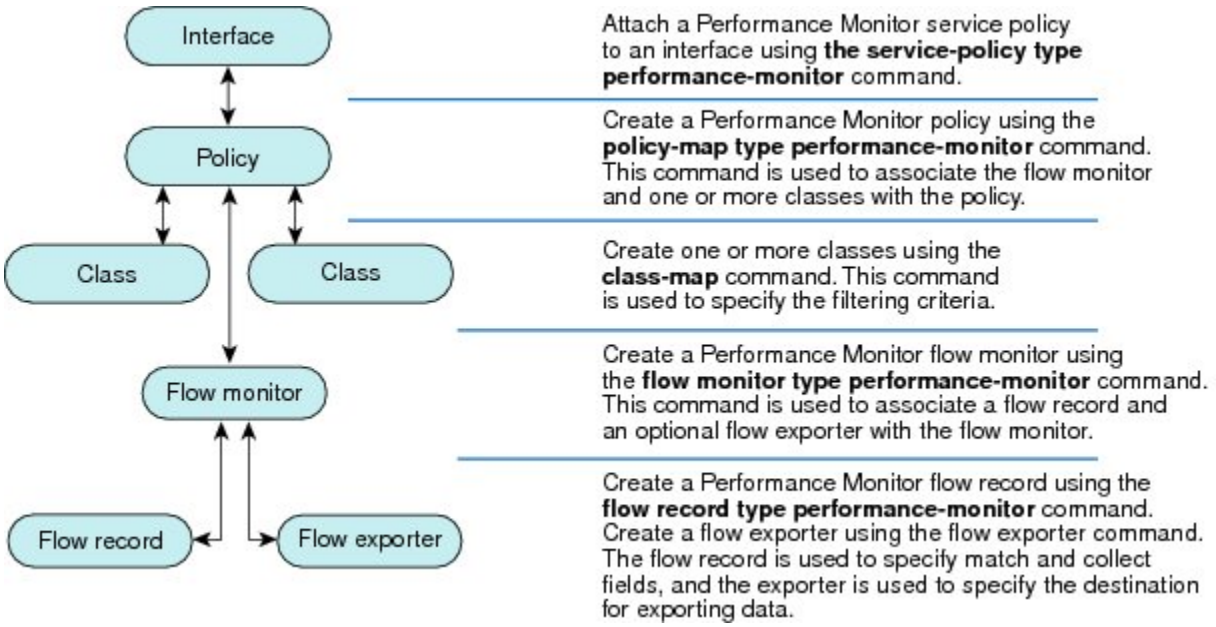
# Configuration Components of Cisco Performance Monitor

To configure Cisco Performance Monitor, configure many of the same basic elements that you normally configure for Flexible NetFlow:

- Interface
- Policy
- Class
- Flow monitor
- Flow record
- Flow exporter

The figure below shows how these elements are related to each other. The elements at the bottom of the figure are configured first.

*Figure 1: Cisco Performance Monitor Components*




As shown above, a policy includes one or more classes. Each class has a flow monitor associated with it, and each flow monitor has a flow record and an optional flow exporter associated with it. These elements are configured in the following order:

1. Configure a flow record to specify the key and non-key fields that you want to monitor. This is configured using **match** and **collect** commands. You can also optimally configure a flow exporter to specify the export destination. For Cisco Performance Monitor, you must configure a **performance-monitor** type flow record.
2. Configure a flow monitor that includes the flow record and flow exporter. For Cisco Performance Monitor, you must configure a **performance-monitor** type flow monitor.
3. Configure a class to specify the filtering criteria using the **class-map** command.
4. Configure a policy to include one or more classes and one or more **performance-monitor** type flow monitors using the **policy-map** command. For Cisco Performance Monitor, you must configure **performance-monitor** type policies.
5. Associate a **performance-monitor** type policy to the appropriate interface using the **service-policy type performance-monitor** command. From release 15.5(2)T, you can add upto three different input and three different output policies on the same interface.

# Data That You Can Monitor Using Cisco Performance Monitor

You can monitor the following information by configuring a flow record with **collect** or **match** commands for the corresponding non-key fields:

**Tip** For more information about these statistics, see the **show performance monitor status**command in the*Cisco Media Monitoring Command Reference.*

- IP Packet Count
- IP TTL
- IP TTL minimum
- IP TTL maximum
- Flow to Interface Mapping
- IP Flow destination address and port, source address and port, and protocol
- RTP Synchronization Source (SSRC)
- IP Octets Count
- Media Stream Packet Count
- Media Stream Octect Count
- Media Byte Rate
- Media Byte Count
- Media Packet Rate
- Media Packet Loss Count
- Media Packet Loss Rate
- Packets Expected Count
- Measured Rate
- Media Loss Event Count
- Round Trip Time (RTT)
- Interarrival Jitter (RFC3550) max
- Interarrival Jitter (RFC3550) min 2
- Interarrival Jitter (RFC3550) mean
- Media Rate Variation
- Monitor Event
- Media Error
- Media Stop
- IP Byte Count
- IP Byte Rate
- IP Source Mask
- IP Destination Mask
- Epoch of A Monitoring Interval
- Packet Forwarding Status

- Packet Drops
- DSCP and IPv6 Traffic Class

# SNMP MIB Support for Cisco Performance Monitor

Cisco Performance Monitor provides support for the use of the industry-standard Simple Network Management Protocol (SNMP) to monitor media streams. This support is implemented with the addition of the following Cisco proprietary SNMP Management Information Base (MIB) modules:

- CISCO-FLOW-MONITOR-TC-MIB—Defines the textual conventions common to the following MIB modules.
- CISCO-FLOW-MONITOR-MIB—Defines the framework that describes the flow monitors supported by a system, the flows that it has learned, and the flow metrics collected for those flows.
- CISCO-RTP-METRICS-MIB—Defines objects that describe the quality metrics collected for RTP streams, similar to those described by an RTCP Receiver Report packet (RFC 3550).
- CISCO-IP-CBR-METRICS-MIB—Defines objects that describe the quality metrics collected for IP streams that have a Constant Bit Rate (CBR).

For detailed information about these MIBs, and to locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at http://www.cisco.com/go/mibs .

This feature also includes two new command-line interface (CLI) commands and one modified CLI command. The commands are as follows:

- **snmp-server host**—Enables the delivery of flow monitoring SNMP notifications to a recipient.
- **snmp-server enable traps flowmon**—Enables flow monitoring SNMP notifications. By default, flow monitoring SNMP notifications are disabled.
- **snmp mib flowmon alarm history**—Sets the maximum number of entries maintained by the flow monitor alarm history log.

For more information about these commands, see the Cisco IOS Master Command List .

# Limitations for the Catalyst 6500 Platform

Cisco Performance Monitor has the following limitations on the Catalyst 6000 platform:

- There are some limitations on which types of interfaces can be monitored. The next two tables list which types of interfaces are supported for ingress and egress monitoring on the Catalyst 6500 platform.

***Table 2: Support for Ingress Interfaces***

| Interface Type | Support |
|---|---|
| Layer 3 Routed Port | Yes |
| Layer 3 Sub-interface (a) | No |
| Layer 3 port channels | Yes |

| Interface Type | Support |
|---|---|
| Layer 3 port-channel sub-interface (a) | No |
| Layer 3 SVI (b) | Partial (see the third bullet below) |
| L3 Tunnels | No |
| Layer 2 Physical (Switched) Ports | Yes |
| Layer 2 Port-channels | Yes |
| Layer 2 Vlans | Yes |

***Table 3: Support for Egress Interfaces***

| Interface Type | Support |
|---|---|
| Layer 3 Routed Port | Yes |
| Layer 3 Sub-interface (a) | Yes |
| Layer 3 port channels | Yes |
| Layer 3 port-channel sub-interface (a) | Yes |
| Layer 3 SVI (b) | Yes |
| L3 Tunnels | No |
| Layer 2 Physical (Switched) Ports | No |
| Layer 2 Port-channels | No |
| Layer 2 Vlans | Yes |

- Performance monitoring on VRFs is not supported.
- Performance monitoring of multicast flows is supported on the ingress direction.
- Routed traffic from a trunk port on a VLAN interface cannot not be monitored because it is not possible to identify the source VLAN interface for the traffic. You will see the following syslog message: “Routed traffic from trunk ports will not be monitored by ingress policy on VLAN interface.”

  For a workaround, you can configure a performance monitoring policy on a trunk interface. This monitoring will result in additional CPU usage.
- You cannot use match all type Class maps. Only match any type of lookups are supported. If you configure performance monitoring to use match-all type class maps, it will result in the cloning of packet to the CPU. Packets will then again be classified in the CPU when match-all classes are properly applied and packet are dropped if required. This causes higher than expected CPU usage.
- Performance monitoring policy on the egress of a VLAN interface will not monitor traffic getting bridged within the VLAN. This is due to hardware limitation. Workaround is to apply the policy at the ingress of VLAN interface as well as egress. Policy on the ingress of the VLAN interface will monitor bridged packets.

- Cloned packets from Egress policies can only be software rate-limited. No hardware-based protection is available for these packets. Therefore, you might see high interrupt CPU usage during scenarios when many flows are being monitored.
- Egress performance monitoring makes use of a recirculation mechanism on the Catalyst 6500 platform. This introduces several microseconds of additional latency to the frame switching.
- Performance monitoring is not supported for the packets switched using the Fast (CEF) Path.
- Lawful intercept and performance monitoring makes use of the same mechanism for cloning the packets. The Lawful Intercept feature takes precedence over performance monitoring. Therefore, performance monitoring does not function when the Lawful Intercept feature is enabled. When this occurs, a syslog message is created.
- Performance monitoring makes use of same mechanism as other features, such as Optimized ACL logging, VACL Capture, IPv6 Copy, and so on. The feature that is enabled first takes precedence. The other features are blocked from being configured and a syslog message is created.

  When reacts (including media-stop) are configured under a performance monitoring policy and when the traffic is unstable, syslog messages are logged into the buffer and are not printed on the console screen.

# Restrictions for Performance Monitor

- On Cisco ASR 1000 Series Aggregation Services Routers, you can configure only 30 fields in a flow record.

# How to Configure Troubleshoot and Maintain Cisco Performance Monitor

**Note** Many of the Flexible NetFlow commands, keywords, and arguments used in used in these tasks are available in previous releases. For more information about these existing Flexible NetFlow commands, keywords, and arguments, refer to the *Cisco IOS Flexible NetFlow Command Reference*.

## Configuring a Flow Exporter for Cisco Performance Monitor

Flow exporters are used to send the data that you collect with Cisco Performance Monitor to a remote system such as a NetFlow Collection Engine. Flow exporters use user datagram protocol (UDP) as the transport protocol and use the Version 9 export format.

To configure a flow exporter for the flow monitor, in order to export the data that is collected by Cisco Performance Monitor to a remote system for further analysis and storage, perform the following optional task. For Cisco Performance Monitor, flow exporters are configured the same way as they are configured for Cisco IOS Flexible NetFlow. For more information. see *Configuring Data Export for Cisco IOS Flexible NetFlow with Flow Exporters*.

**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.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow exporter** *exporter-name*
4. **description** *description*
5. **destination** {*ip-address* | *hostname*} [**vrf** *vrf-name*]
6. **export-protocol netflow-v9**
7. **dscp** *dscp*
8. **source** *interface-type interface-number*
9. **option** {**exporter-stats** | **interface-table** | **sampler-table**} [**timeout** *seconds*]
10. **output-features**
11. **template data timeout** *seconds*
12. **transport udp** *udp-port*
13. **ttl** *seconds*
14. **end**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **flow exporter** *exporter-name*<br>**Example:**<br>`Router(config)# flow exporter EXPORTER-1` | Creates the flow exporter and enters Flexible NetFlow flow exporter configuration mode.<br>• This command also allows you to modify an existing flow exporter. |
| **Step 4** | **description** *description*<br>**Example:**<br>`Router(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*]<br>**Example:**<br>`Router(config-flow-exporter)# destination 172.16.10.2` | Specifies the IP address or hostname of the system to which the exporter sends data. |
| **Step 6** | **export-protocol netflow-v9**<br>**Example:**<br>`Router(config-flow-exporter)# export-protocol netflow-v9` | Specifies the version of the NetFlow export protocol used by the exporter. Only the default value (**netflow-v9)** is supported. |
| **Step 7** | **dscp** *dscp*<br>**Example:**<br>`Router(config-flow-exporter)# dscp 63` | (Optional) Configures differentiated services code point (DSCP) parameters for datagrams sent by the exporter.<br>• The range for the *dscp* argument is from 0 to 63. Default: 0. |
| **Step 8** | **source** *interface-type interface-number*<br>**Example:**<br>`Router(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 9** | **option** {**exporter-stats** \| **interface-table** \| **sampler-table**} [**timeout** *seconds*]<br>**Example:**<br>`Router(config-flow-exporter)# option exporter-stats timeout 120` | (Optional) Configures options data parameters for the exporter.<br>• You can configure all three options concurrently.<br>• The range for the *seconds* argument is 1 to 86,400. Default: 600. |
| **Step 10** | **output-features**<br>**Example:**<br>`Router(config-flow-exporter)# output-features` | (Optional) Enables sending export packets using quality of service (QoS) and encryption. |
| **Step 11** | **template data timeout** *seconds*<br>**Example:**<br>`Router(config-flow-exporter)# template data timeout 120` | (Optional) Configure resending of templates based on a timeout.<br>• The range for the *seconds* argument is 1 to 86400 (86400 seconds = 24 hours). |
| **Step 12** | **transport udp** *udp-port*<br>**Example:**<br>`Router(config-flow-exporter)# transport udp 650` | Configures UDP as the transport protocol and specifies the UDP port on which the destination system is listening for exported datagrams.<br>• The range for the *udp-port* argument is from 1 to 65536. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 13** | **ttl** *seconds*<br>**Example:**<br>`Router(config-flow-exporter)# ttl 15` | (Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter.<br>• The range for the *seconds* argument is from 1 to 255. |
| **Step 14** | **end**<br>**Example:**<br>`Router(config-flow-exporter)# end` | Exits flow exporter configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the configuration and status of your flow exporter, use the **show flow exporter** command.

# Configuring a Flow Record for Cisco Performance Monitor

The basic concepts and techniques for configuring a flow record for Cisco Performance Monitor are the same as flow records for Flexible NetFlow. The flow record specifies how the data collected data is aggregated and presented. The only significant difference is that, for Cisco Performance Monitor, the command includes **type performance-monitor**.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record type performance-monitor** *record-name*
4. **match ipv4** {**destination**{**address** | **prefix**[**minimum-mask** *mask*]}| **protocol**| **source**{**address** | **prefix**[**minimum-mask** *mask*]}
5. **match transport** {**destination-port**| **rtp** [**ssrc**]| **source-port**}
6. **collect application media** {**bytes**{**rate**| **counter**}| **packets**{**rate**| **counter**}| **events**}
7. **collect counter** {**bytes**[**long**| **rate**]| **packets**[**dropped**[**long**]| **long**]}
8. **collect interface** {**input**| **output**}
9. **collect ipv4** {**destination mask**[**minimum-mask** *mask*]}| **dscp**| **source mask**[**minimum-mask** *mask*] | **ttl**[**minimum** | **maximum**]}
10. **collect monitor event**
11. **collect routing forwarding-status** [**reason**]
12. **collect timestamp internal**
13. **collect transport** {**event packet-loss counter** | **packets**{**expected counter**| **lost**{**counter**| **rate**}}| **round-trip-time**| **rtp jitter**{**minimum**| **mean**| **maximum**}}
14. **collect flow direction**
15. **end**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **flow record type performance-monitor** *record-name*<br>**Example:**<br>`Router(config)# flow record type performance-monitor record-8` | Creates a flow record and enters flow record configuration mode.<br>• This command also allows you to modify an existing flow record. |
| **Step 4** | **match ipv4** {**destination**{**address** \| **prefix**[**minimum-mask** *mask*]}\| **protocol**\| **source**{**address** \| **prefix**[**minimum-mask** *mask*]} <br>**Example:**<br>`Router(config-flow-record)# match ipv4 destination address` | Specifies that one or more of the IPv4 fields will be used as a key field. |
| **Step 5** | **match transport** {**destination-port**\| **rtp** [**ssrc**]\| **source-port**}<br>**Example:**<br>`Router(config-flow-record)# match transport destination-port` | Specifies that one or more of the transport layer fields will be used as a key field, including the Synchronization Source (SSRC) field in the Real-Time Transport Protocol (RTP) packet header. |
| **Step 6** | **collect application media** {**bytes**{**rate**\| **counter**}\| **packets**{**rate**\| **counter**}\| **events**}<br>**Example:**<br>`Router(config-flow-record)# collect application media events` | Specifies that the application media bytes, packets, or events will be used as a nonkey field. An application event occurs when either one of the thresholds specified by a react statement for the flow was crossed at least once in the monitoring interval or no media packets were seen. |
| **Step 7** | **collect counter** {**bytes**[**long**\| **rate**]\| **packets**[**dropped**[**long**]\| **long**]}<br>**Example:**<br>`Router(config-flow-record)# collect counter bytes long` | Specifies the number of bytes or packets that will be used as a nonkey field. |
| **Step 8** | **collect interface** {**input**\| **output**}<br>**Example:** | Specifies that the input or output interface will be used as a nonkey field. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router(config-flow-record)# collect interface input` | |
| **Step 9** | **collect ipv4** {**destination mask**[**minimum-mask** *mask*]}\| **dscp**\| **source mask**[**minimum-mask** *mask*] \| **ttl**[**minimum** \| **maximum**]}<br>**Example:**<br>`Router(config-flow-record)# collect ipv4 dscp` | Specifies that the IPv4 differentiated services code point (DCSP) field or the IPv4 time-to-live (TTL) field will be used as a nonkey field. |
| **Step 10** | **collect monitor event**<br>**Example:**<br>`Router(config-flow-record)# collect monitor event` | Specifies that the monitor event field will be used as a nonkey field. A monitor event occurs when no media application packets were seen |
| **Step 11** | **collect routing forwarding-status** [**reason**]<br>**Example:**<br>`Router(config-flow-record)# collect routing forwarding-status` | Specifies that the one or more of the routing attributes will be used as a nonkey field. |
| **Step 12** | **collect timestamp internal**<br>**Example:**<br>`Router(config-flow-record)# collect timestamp internal` | Specifies that the system timestamp of the first seen or last seen packet in a flow will be used as a nonkey field. |
| **Step 13** | **collect transport** {**event packet-loss counter** \| **packets**{**expected counter**\| **lost**{**counter**\| **rate**}}\| **round-trip-time**\| **rtp jitter**{**minimum**\| **mean**\| **maximum**}}<br>**Example:**<br>`Router(config-flow-record)# collect transport packets expected counter` | Specifies that one or more of the transport layer fields will be used as a nonkey field. These fields include metrics for:<br>• Packet-loss counter<br>• Expected packets counter<br>• Jitter |
| **Step 14** | **collect flow direction**<br>**Example:**<br>`Router(config-flow-record)# collect flow direction` | Specifies that the flow direction field will be used as a nonkey field. |
| **Step 15** | **end**<br>**Example:**<br>`Router(config-flow-record)# end` | Exits flow record configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the configuration and status of your flow record, use the **show flow record type performance-monitor** command.

# Configuring a Flow Monitor for Cisco Performance Monitor

The basic concepts for configuring a flow monitor for Cisco Performance Monitor are the same as flow monitors for Flexible NetFlow. Each flow monitor has a separate cache assigned to it and requires a record to define the contents and layout of its cache entries.

When you configure a flow monitor, you must use either:

- An existing flow record that you configured
- One of the following default predefined records:
  - The default RTP record (**default-rtp**)
  - The default TCP record (**default-tcp**)
  - Flexible NetFlow's "NetFlow IPv4 original input"

**Note** To modify a flow record, you must remove it from all flow monitors it is associated with.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor type performance-monitor** *monitor-name*
4. **description** *description*
5. **cache** {**entries**| **timeout**| **type**}
6. **statistics** {**packet**}
7. **exporter** *exporter-name*
8. **record** {*record-name*| **default-rtp**| **default-tcp**|**netflow ipv4 original-input**}
9. **end**

### DETAILED STEPS

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

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **flow monitor type performance-monitor** *monitor-name*<br>**Example:**<br>`Device(config)# flow monitor type performance-monitor FLOW-MONITOR-2` | Creates a flow monitor and enters flow monitor configuration mode.<br>• This command also allows you to modify an existing flow monitor. |
| **Step 4** | **description** *description*<br>**Example:**<br>`Device(config-flow-monitor)# description Used for monitoring IPv4 traffic` | (Optional) Creates a description for the flow monitor. |
| **Step 5** | **cache** {**entries**\| **timeout**\| **type**}<br>**Example:**<br>`Device(config-flow-monitor)# cache timeout 20` | (Optional) Creates a cache for the flow monitor. |
| **Step 6** | **statistics** {**packet**}<br>**Example:**<br>`Device(config-flow-monitor)# statistics` | (Optional) specifies whether statistics are collected for the flow monitor. |
| **Step 7** | **exporter** *exporter-name*<br>**Example:**<br>`Device(config-flow-monitor)# exporter export-4` | Specifies the flow exporter for the flow monitor. |
| **Step 8** | **record** {*record-name*\| **default-rtp**\| **default-tcp**\|**netflow ipv4 original-input**}<br>**Example:**<br>`Device(config-flow-monitor)# record default-rtp` | Specifies the flow record for the flow monitor. |
| **Step 9** | **end**<br>**Example:**<br>`Device(config-flow-monitor)# end` | Exits flow monitor configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the configuration and status of your flow monitor, use the **show flow monitor type performance-monitor** command and the **show running-config flow monitor** command.

# Configuring a Flow Class for Cisco Performance Monitor

The basic concepts and techniques for configuring a class for Cisco Performance Monitor are the same as for any other type of class. The class specifies the filter that determines which flow traffic to monitor. The filter is configured using various match commands in class-map mode.

If you do not already have a flow monitor configured, see Configuring a Flow Monitor for Cisco Performance Monitor, on page 49:

**Note** Nested class maps are not supported. In other words, you cannot use the **class-map** command while in class-map configuration mode (config-cmap).

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** *class-name*
4. **description** *description*
5. **match** {*access-group* {*access-group* | **name** *access-group-name*} | **any** | **class-map** *class-map-name* | **cos** *cos-value* | **destination-address mac** *address* | **discard-class** *class-number* | **dscp** *dscp-value* | **flow** {**direction** | **sampler**} | **fr-de** | **fr-dlci** *dlci-number* | **input-interface** *interface-name* | **ip** {**rtp** *starting-port-number port-range* | **precedence** | **dscp**} | **mpls experimental topmost** *number* | **not** *match-criterion*| **packet length** {**max** *maximum-length-value* [**min** *minimum-length-value*] | **min** *minimum-length-value* [**max** *maximum-length-value*]} | **precedence** {*precedence-criteria1* | *precedence-criteria2* | *precedence-criteria3* | *precedence-criteria4*} | **protocol** *protocol-name* | **qos-group** *qos-group-value* | **source-address** *mac address-destination*| **vlan** {*vlan-id* | *vlan-range* | *vlan-combination*}}
6. **rename** *class-name*
7. **end**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **class-map** *class-name*<br>**Example:**<br>`Device(config)# class-map class-4` | Specifies a class to include in the policy. Repeat this command for each class that you want to include in the policy. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | **description** *description*<br>**Example:**<br>`Device(config-cmap)# description match any packets` | (Optional) Creates a description for the flow class. |
| **Step 5** | **match** {*access-group* {*access-group* \| **name** *access-group-name*} \| **any** \| **class-map** *class-map-name* \| **cos** *cos-value* \| **destination-address mac** *address* \| **discard-class** *class-number* \| **dscp** *dscp-value* \| **flow** {**direction** \| **sampler**} \| **fr-de** \| **fr-dlci** *dlci-number* \| **input-interface** *interface-name* \| **ip** {**rtp** *starting-port-number port-range* \| **precedence** \| **dscp**} \| **mpls experimental topmost** *number* \| **not** *match-criterion*\| **packet length** {**max** *maximum-length-value* [**min** *minimum-length-value*] \| **min** *minimum-length-value* [**max** *maximum-length-value*]} \| **precedence** {*precedence-criteria1* \| *precedence-criteria2* \| *precedence-criteria3* \| *precedence-criteria4*} \| **protocol** *protocol-name* \| **qos-group** *qos-group-value* \| **source-address** *mac address-destination*\| **vlan** {*vlan-id* \| *vlan-range* \| *vlan-combination*}}<br>**Example:**<br>`Device(config-cmap)# match any` | Specifies the classification criteria.<br>For more information and examples, see the *Cisco Media Monitoring Command Reference*. |
| **Step 6** | **rename** *class-name*<br>**Example:**<br>`Device(config-cmap)# rename class-4` | Specifies a new name for the flow class. |
| **Step 7** | **end**<br>**Example:**<br>`Device(config-cmap)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the configuration and status of your flow class, use the **show policy-map type performance-monitor** or **show class-map** command.

# Configuring a Flow Policy for Cisco Performance Monitor Using an Existing Flow Monitor

The basic concepts and techniques for configuring a class for Cisco Performance Monitor are the same as for any other type of class. The class specifies which flow monitor is included. The only significant difference is that, for Cisco Performance Monitor, the **policy-map** command includes **type performance-monitor**.

If you do not already have a flow monitor configured or do not want to use any of your existing flow monitors for a new class, you can configure it using the flow monitor inline option and specifying which flow record and flow exporter are included.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map type performance-monitor** *policy-name*
4. **parameter-map type performance-monitor system-default-aor**
5. **class** {*class-name* | **class-default**}
6. **flow monitor** *monitor-name*
7. **monitor metric ip-cbr**
8. **rate layer3** {byte-*rate* {**bps** | **kbps** | **mbps** | **gbps**} | **packet**}
9. **exit**
10. **monitor metric rtp**
11. **clock-rate** {*type-number* | *type-name* | **default**} *rate*
12. **max-dropout** *number*
13. **max-reorder** *number*
14. **min-sequential** *number*
15. **ssrc maximum** *number*
16. exit
17. **monitor parameters**
18. **flows** *number*
19. **interval duration** *number*
20. **history** *number*
21. **timeout** *number*
22. **exit**
23. **react** *ID* {**media-stop** | **mrv** | **rtp-jitter-average** | **transport-packets-lost-rate**}
24. **action** {**snmp** | **syslog**}
25. **alarm severity** {**alert** | **critical** | **emergency** | **error** | **info**}
26. **alarm type** {**discrete** | **grouped {count** *number* | **percent** *number*}
27. **threshold value** {**ge** *number* | **gt** *number* | **le** *number* | **lt** *number* | **range** *rng-start rng-end*}
28. **description** *description*
29. **end**

## DETAILED STEPS

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

| | Command or Action | Purpose |
|---|---|---|
| | `Device# configure terminal` | |
| **Step 3** | **policy-map type performance-monitor** *policy-name*<br>**Example:**<br>`Device(config)# policy-map type performance-monitor FLOW-MONITOR-4` | Creates a policy and enters policy configuration mode.<br>• This command also allows you to modify an existing policy. |
| **Step 4** | **parameter-map type performance-monitor system-default-aor**<br>**Example:**<br>`Device(config-pmap)# parameter-map type performance-monitor system-default-aor` | Creates a parameter map for Performance Monitor. The only map available is the system-default -aor map |
| **Step 5** | **class** {*class-name* \| **class-default**}<br>**Example:**<br>`Device(config-pmap)# class class-4` | Specifies a class to include in the policy. Repeat this command for each class that you want to include in the policy. |
| **Step 6** | **flow monitor** *monitor-name*<br>**Example:**<br>`Device(config-pmap-c)# flow monitor FLOW-MONITOR-4` | Enters flow monitor configuration mode. If you do not want to use an existing flow monitor, you can use the **inline** option to configure a new one, as described in the Configuring a Flow Policy for Cisco Performance Monitor Without Using an Existing Flow Monitor, on page 57. |
| **Step 7** | **monitor metric ip-cbr**<br>**Example:**<br>`Device(config-pmap-c)# monitor metric ip-cbr` | (Optional) Enters IP-CBR monitor metric configuration mode. |
| **Step 8** | **rate layer3** {byte-*rate* {**bps** \| **kbps** \| **mbps** \| **gbps**} \| **packet**}<br>**Example:**<br>`Device(config-pmap-c-mipcbr)# rate layer3 248 mbps` | (Optional) Specifies the rate for monitoring the metrics.<br>• *byte-rate* --Data rate in Bps, kBps, mBps, or gBps. The range is 1 to 65535.<br>• **packet** --Packet rate in packets per second. |
| **Step 9** | **exit**<br>**Example:**<br>`Device(config-pmap-c-mipcbr)# exit` | Returns to policy class configuration mode. |
| **Step 10** | **monitor metric rtp**<br>**Example:**<br>`Device(config-pmap-c)# monitor metric rtp` | Enters RTP monitor metric configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 11** | **clock-rate** {*type-number* \| *type-name* \| **default**} *rate*<br>**Example:**<br>`Device(config-pmap-c-mrtp)# clock-rate 8 9600` | Specifies the clock rate used to sample RTP video-monitoring metrics.<br>For more information about the clock-type numbers and names, see the *Cisco Media Monitoring Command Reference.*<br>The range for *rate* is 1 kHz to 192 kHz. |
| **Step 12** | **max-dropout** *number*<br>**Example:**<br>`Device(config-pmap-c-mrtp)# max-dropout 2` | Specifies the maximum number of dropouts allowed when sampling RTP video-monitoring metrics. |
| **Step 13** | **max-reorder** *number*<br>**Example:**<br>`Device(config-pmap-c-mrtp)# max-reorder 4` | Specifies the maximum number of reorders allowed when sampling RTP video-monitoring metrics. |
| **Step 14** | **min-sequential** *number*<br>**Example:**<br>`Device(config-pmap-c-mrtp)# min-sequential 2` | Specifies the minimum number of sequential packets required to identify a stream as being an RTP flow. |
| **Step 15** | **ssrc maximum** *number*<br>**Example:**<br>`Device(config-pmap-c-mrtp)# ssrc maximum 20` | Specifies the maximum number of SSRCs that can be monitored within the same flow. A flow is defined by the protocol, source/destination address, and source/destination port). |
| **Step 16** | exit<br>**Example:**<br>`Device(config-pmap-c-mrtp)# exit` | Returns to policy class configuration mode. |
| **Step 17** | **monitor parameters**<br>**Example:**<br>`Device(config-pmap-c)# monitor parameters` | Enters monitor parameters configuration mode. |
| **Step 18** | **flows** *number*<br>**Example:**<br>`Device(config-pmap-c-mparam)# flows 40` | Specifies the maximum number of flows for each monitor cache. |
| **Step 19** | **interval duration** *number*<br>**Example:**<br>`Device(config-pmap-c-mparam)# interval duration 40` | Specifies the interval, in seconds, between samples taken of video-monitoring metrics. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 20** | **history** *number*<br>**Example:**<br>`Device(config-pmap-c-mparam)# history 4` | Specifies the number of historical buckets of collected video-monitoring metrics. |
| **Step 21** | **timeout** *number*<br>**Example:**<br>`Device(config-pmap-c-mparam)# timeout 20` | Specifies the number of intervals before a stopped flow is removed from the database. |
| **Step 22** | **exit**<br>**Example:**<br>`Device(config-pmap-c-mparam)# exit` | Returns to policy class configuration mode. |
| **Step 23** | **react** *ID* {**media-stop** \| **mrv** \| **rtp-jitter-average** \| **transport-packets-lost-rate**}<br>**Example:**<br>`Device(config-pmap-c)# react 41 rtp-jitter-average` | Enters a mode where you can specify what reaction occurs when a threshold is violated for the following metrics:<br>• *ID*-- ID for react configuration. Range is 1 to 65535.<br>• **media-stop** --No traffic is found for the flow.<br>• **mrv** --Ratio calculated by dividing the difference between the actual rate and the expected rate, by the expected rate.<br>• **rtp-jitter-average** --Average jitter.<br>• **transport-packets-lost-rate** --Ratio calculated by dividing the number of lost packets by the expected packet count. |
| **Step 24** | **action** {**snmp** \| **syslog**}<br>**Example:**<br>`Device(config-pmap-c-react)# action syslog` | Specifies how violations of the thresholds with be reported. |
| **Step 25** | **alarm severity** {**alert** \| **critical** \| **emergency** \| **error** \| **info**}<br>**Example:**<br>`Device(config-pmap-c-react)# alarm severity critical` | Specifies which level of alarm will be reported. The default setting is **info**. |
| **Step 26** | **alarm type** {**discrete** \| **grouped** {**count** *number* \| **percent** *number*}<br>**Example:**<br>`Device(config-pmap-c-react)# alarm type discrete` | Specifies which types of levels are considered alarms that require reporting. The default setting is **discrete**. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 27** | **threshold value** {**ge** *number* \| **gt** *number* \| **le** *number* \| **lt** *number* \| **range** *rng-start rng-end*}<br>**Example:**<br>`Device(config-pmap-c-react)# threshold value ge 20` | Specifies which types of threshold values are considered alarms that require reporting.<br>If no value is set but the application name is configured as a key field, then the system uses the value for the threshold that it finds in the default map. If no value is set and the application name is not configured as a key field, then the default value is used for the threshold.<br>If more than one react command is configured for the same policy and class but only one of the react configurations has threshold values set, then the values of the configured react take precedence and the rest of the threshold values are ignored.<br>If more than one react command is configured for the same policy and none of them have the threshold value configured, then the default threshold value is applied for the configuration with the lowest react ID. |
| **Step 28** | **description** *description*<br>**Example:**<br>`Device(config-cmap-c-react)# description rtp-jitter-average above 40` | (Optional) Creates a description for the reaction. |
| **Step 29** | **end**<br>**Example:**<br>`Device(config-pmap-c-react)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the configuration and status of your flow policy, use the **show policy-map type performance-monitor** command.

# Configuring a Flow Policy for Cisco Performance Monitor Without Using an Existing Flow Monitor

The basic concepts and techniques for configuring a class for Cisco Performance Monitor are the same as for any other type of class. The class specifies which flow monitor is included. The only significant difference is that, for Cisco Performance Monitor, the **policy-map** command includes **type performance-monitor**.

If you do not already have a flow monitor configured or do not want to use any of your existing flow monitors for a new class, you can configure it under the class configuration mode, by specifying which flow record and flow exporter are included.

### SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **policy-map type performance-monitor** *policy-name*
4. **class** {*class-name* | **class-default**}
5. **flow monitor inline**
6. **record** {*record-name*| **default-rtp**| **default-tcp**}
7. **exporter** *exporter-name*
8. **exit**
9. monitor metric ip-cbr
10. **rate layer3** {*byte-rate* {**bps** | **kbps** | **mbps** | **gbps**} | **packet**}
11. **exit**
12. **monitor metric rtp**
13. **clock-rate** {*type-number*| *type-name*} *rate*
14. **max-dropout** *number*
15. **max-reorder** *number*
16. **min-sequential** *number*
17. **ssrc maximum** *number*
18. exit
19. **monitor parameters**
20. **flows** *number*
21. **interval duration** *number*
22. **history** *number*
23. **timeout** *number*
24. **exit**
25. **react** *ID* {**media-stop** | **mrv** | **rtp-jitter-average** | **transport-packets-lost-rate**}
26. **action** {**snmp** | **syslog**}
27. **alarm severity** {**alert**| **critical**| **emergency**| **error**| **info**}
28. **alarm type** {**discrete**| **grouped** {**count** *number* | **percent** *number*}
29. **threshold value** {**ge** *number* | **gt** *number* | **le** *number* | **lt** *number* | **range** *rng-start rng-end*
30. **description** *description*
31. **end**

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **policy-map type performance-monitor** *policy-name* | Creates a policy and enters policy configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config)# policy-map type preformance-monitor FLOW-MONITOR-4` | • This command also allows you to modify an existing policy. |
| **Step 4** | **class** {*class-name* \| **class-default**}<br>**Example:**<br>`Router(config-pmap)# class class-4` | Specifies a class to include in the policy. Repeat this command for each class that you want to include in the policy. |
| **Step 5** | **flow monitor inline**<br>**Example:**<br>`Router(config-pmap-c)# flow monitor inline` | Enters inline mode and enables you to configure a new flow monitor. |
| **Step 6** | **record** {*record-name*\| **default-rtp**\| **default-tcp**}<br>**Example:**<br>`Router(config-pmap-c-flowmon)# record default-tcp` | Specifies a flow record to associate with the flow monitor. |
| **Step 7** | **exporter** *exporter-name*<br>**Example:**<br>`Router(config-pmap-c-flowmon)# exporter exporter-4` | Specifies a flow record to associate with the flow exporter. |
| **Step 8** | **exit**<br>**Example:**<br>`Router(config-pmap-c-flowmon)# exit` | Returns to policy class configuration mode. |
| **Step 9** | monitor metric ip-cbr<br>**Example:**<br>`Router(config-pmap-c)# monitor metric ip-cbr` | (Optional) Enters IP-CBR monitor metric configuration mode. |
| **Step 10** | **rate layer3** {*byte-rate* {**bps** \| **kbps** \| **mbps** \| **gbps**} \| **packet**}<br>**Example:**<br>`Router(config-pmap-c-mipcbr)# rate layer3 248 mbps` | (Optional) Specifies the rate for monitoring the metrics.<br>• *byte-rate* --Data rate in Bps, kBps, mBps, or gBps. The range is 1 to 65535.<br>• **packet** --Packet rate in packets per second. |
| **Step 11** | **exit**<br>**Example:**<br>`Router(config-pmap-c-mipcbr)# exit` | Returns to policy class configuration mode. |
| **Step 12** | **monitor metric rtp**<br>**Example:** | Enters RTP monitor metric configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router(config-pmap-c)# monitor metric rtp` | |
| **Step 13** | **clock-rate** {*type-number*\| *type-name*} *rate*<br>**Example:**<br>`Router(config-pmap-c-mrtp)# clock-rate 8 9600` | Specifies the clock rate used to sample RTP video-monitoring metrics.<br>For more information about the clock-type numbers and names, see the *Cisco Media Monitoring Command Reference*.<br>The range for *rate* is 1 kHz to 192 kHz. |
| **Step 14** | **max-dropout** *number*<br>**Example:**<br>`Router(config-pmap-c-mrtp)# max-dropout 2` | Specifies the maximum number of dropouts allowed when sampling RTP video-monitoring metrics. |
| **Step 15** | **max-reorder** *number*<br>**Example:**<br>`Router(config-pmap-c-mrtp)# max-reorder 4` | Specifies the maximum number of reorders allowed when sampling RTP video-monitoring metrics. |
| **Step 16** | **min-sequential** *number*<br>**Example:**<br>`Router(config-pmap-c-mrtp)# min-sequential 2` | Specifies the minimum number of sequental packets required to identify a stream as being an RTP flow. |
| **Step 17** | **ssrc maximum** *number*<br>**Example:**<br>`Router(config-pmap-c-mrtp)# ssrc maximum 20` | Specifies the maximum number of SSRCs that can be monitored within the same flow. A flow is defined by the protocol, source/destination address, and source/destination port). |
| **Step 18** | exit<br>**Example:**<br>`Router(config-pmap-c-mrtp)# exit` | Returns to policy class configuration mode. |
| **Step 19** | **monitor parameters**<br>**Example:**<br>`Router(config-pmap-c)# monitor parameters` | Enters monitor parameters configuration mode. |
| **Step 20** | **flows** *number*<br>**Example:**<br>`Router(config-pmap-c-mparam)# flows 40` | Specifies the maximum number of flows for each monitor cache. |
| **Step 21** | **interval duration** *number*<br>**Example:** | Specifies the duration of the intervals, in seconds, for collecting monitoring metrics. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router(config-pmap-c-mparam)# interval duration 40` | |
| **Step 22** | **history** *number*<br>**Example:**<br>`Router(config-pmap-c-mparam)# history 4` | Specifies the number of historical intervals of collected monitoring metrics to display. |
| **Step 23** | **timeout** *number*<br>**Example:**<br>`Router(config-pmap-c-mparam)# timeout 20` | Specifies the number intervals before a stopped flow is removed from the database. |
| **Step 24** | **exit**<br>**Example:**<br>`Router(config-pmap-c-mparam)# exit` | Returns to policy class configuration mode. |
| **Step 25** | **react** *ID* {**media-stop** \| **mrv** \| **rtp-jitter-average** \| **transport-packets-lost-rate**}<br>**Example:**<br>`Router(config-pmap-c)# react 41 rtp-jitter-average` | Enters a mode where you can specify what reaction occurs when a threshold is violated for the following metrics:<br>• *ID*-- ID for react configuration. Range is 1 to 65535.<br>• **media-stop** --No traffic is found for the flow.<br>• **mrv** --Ratio calculated by dividing the difference between the actual rate and the expected rate, by the expected rate.<br>• **rtp-jitter-average** --Average jitter.<br>• **transport-packets-lost-rate** --Ratio calculated by dividing the number of lost packets by the expected packet count. |
| **Step 26** | **action** {**snmp** \| **syslog**}<br>**Example:**<br>`Router(config-pmap-c-react)# action syslog` | Specifies how violations of the thresholds with be reported. |
| **Step 27** | **alarm severity** {**alert**\| **critical**\| **emergency**\| **error**\| **info**}<br>**Example:**<br>`Router(config-pmap-c-react)# alarm severity critical` | Specifies which level of alarm will be reported.The default setting is **info**. |
| **Step 28** | **alarm type** {**discrete**\| **grouped** {**count** *number* \| **percent** *number*}<br>**Example:** | Specifies which types of levels are considered alarms that require reporting. The default setting is **discrete**. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router(config-pmap-c-react)# alarm severity critical` | |
| **Step 29** | **threshold value** {**ge** *number* \| **gt** *number* \| **le** *number* \| **lt** *number* \| **range** *rng-start rng-end*<br>**Example:**<br>`Router(config-pmap-c-react)# threshold value ge` | Specifies which types of levels values are considered alarms that require reporting. |
| **Step 30** | **description** *description*<br>**Example:**<br>`Router(config-cmap-c-react)# description rtp-jitter-average above 40` | (Optional) Creates a description for the reaction. |
| **Step 31** | **end**<br>**Example:**<br>`Router(config-pmap-c-react)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the configuration and status of your flow policy, use the **show policy-map type performance-monitor** command.

# Applying a Cisco Performance Monitor Policy to an Interface Using an Existing Flow Policy

Before it can be activated, a Cisco Performance Monitor policy must be applied to at least one interface. To activate a Cisco Performance Monitor policy, perform the following required task.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **service-policy type performance-monitor** {**input** | **output**} *policy-name*
5. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **interface** *type number*<br>**Example:**<br>`Router(config)# interface ethernet 0/0` | Specifies an interface and enters interface configuration mode. |
| **Step 4** | **service-policy type performance-monitor** {**input** \| **output**} *policy-name*<br>**Example:**<br>`Router(config-if)# service-policy type performance-monitor input mypolicy-map-4`<br>`Router(config-if)# service-policy type performance-monitor input rtp`<br>`Router(config-if)# service-policy type performance-monitor input tcp`<br>**Example:**<br>`Router(config-if)# service-policy type performance-monitor output rtp`<br>`Router(config-if)# service-policy type performance-monitor output tcp`<br>`Router(config-if)# service-policy type performance-monitor output test` | Attaches a policy map to an input interface or virtual circuit (VC), or an output interface or VC, to be used as the service policy for that interface or VC.<br>• **input** --Attaches the specified policy map to the input interface or input VC.<br>• **output** --Attaches the specified policy map to the output interface or output VC.<br>• *policy-name* --name of a service policy map (created by the **policy-map** command) to be attached. The name can be a maximum of 40 alphanumeric characters. |
| **Step 5** | **end**<br>**Example:**<br>`Router(config-if)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

## Troubleshooting Tips

To check the configuration and status of your service policy, use the following commands:

- **show performance monitor history**
- **show performance monitor status**
- **show policy-map ypre performance-monitor interface**

# Applying a Cisco Performance Monitor Policy to an Interface Without Using an Existing Flow Policy

Before it can be activated, a Cisco Performance Monitor policy must be applied to at least one interface. To activate a Cisco Performance Monitor policy, perform the following required task.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **service-policy type performance-monitor inline** {**input** | **output**}
5. **match** {*access-group* {*access-group* | **name** *access-group-name*} | **any** | **class-map**class-map-name | **cos** cos-value | **destination-address mac** address | **discard-class** *class-number* | dscp *dscp-value* | **flow** {**direction** | **sampler**} | **fr-de** | **fr-dlci** *dlci-number* | **input-interface** *interface-name* | **ip** {**rtp** *starting-port-number port-range* | **precedence** | **dscp**} | **mpls experimental topmost** *number* | **not** *match-criterion*| **packet length** {**max** *maximum-length-value* [**min** *minimum-length-value*] | **min** *minimum-length-value* [**max** *maximum-length-value*]} | **precedence** {*precedence-criteria1* | *precedence-criteria2* | *precedence-criteria3* | *precedence-criteria4*} | **protocol** *protocol-name* | **qos-group** *qos-group-value* | **source-address** *mac address-destination*| **vlan** {*vlan-id*| *vlan-range* | *vlan-combination*}}
6. **flow monitor** {*monitor-name*| **inline**}
7. **record** {r*ecord-name*| **default-rtp**| **default-tcp**}
8. **exporter** *exporter-name*
9. exit
10. monitor metric ip-cbr
11. **rate layer3** {*byte-rate* {**bps** | **kbps** | **mbps** | **gbps**} | **packet**}
12. **exit**
13. **monitor metric rtp**
14. **clock-rate** {*type-number*| *type-name*} *rate*
15. **max-dropout** *number*
16. **max-reorder** *number*
17. **min-sequential** *number*
18. **ssrc maximum** *number*
19. exit
20. **monitor parameters**
21. **flows** *number*
22. **interval duration** *number*
23. **history** *number*
24. **timeout** *number*
25. **exit**
26. **react** *ID* {**media-stop** | **mrv** | **rtp-jitter-average** | **transport-packets-lost-rate**}
27. **action** {**snmp** | **syslog**}
28. **alarm severity** {**alert**| **critical**| **emergency**| **error** | **info**}
29. **alarm type** {**discrete**| **grouped**{**count** *number* | **percent** *number*}}
30. **threshold value** {**ge** *number* | **gt** *number* | **le** *number* | **lt** *number* | **range** *rng-start rng-end*}
31. **end**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable** | Enables privileged EXEC mode. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router> enable` | • Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **interface** *type number*<br>**Example:**<br>`Router(config)# interface ethernet 0/0` | Specifies an interface and enters interface configuration mode. |
| **Step 4** | **service-policy type performance-monitor inline** {**input** \| **output**}<br>**Example:**<br>**Example:**<br>`Router(config-if)# service-policy type performance-monitor inline input` | Attaches a policy map to an input interface or virtual circuit (VC), or an output interface or VC, to be used as the service policy for that interface or VC.<br>• **input** --Attaches the specified policy map to the input interface or input VC.<br>• **output** --Attaches the specified policy map to the output interface or output VC. |
| **Step 5** | **match** {*access-group* {*access-group* \| **name** *access-group-name*} \| **any** \| **class-map**class-map-name \| **cos** cos-value \| **destination-address mac** address \| **discard-class** *class-number* \| dscp *dscp-value* \| **flow** {**direction** \| **sampler**} \| **fr-de** \| **fr-dlci** *dlci-number* \| **input-interface** *interface-name* \| **ip** {**rtp** *starting-port-number port-range* \| **precedence** \| **dscp**} \| **mpls experimental topmost** *number* \| **not** *match-criterion*\| **packet length** {**max** *maximum-length-value* [**min** *minimum-length-value*] \| **min** *minimum-length-value* [**max** *maximum-length-value*]} \| **precedence** {*precedence-criteria1* \| *precedence-criteria2* \| *precedence-criteria3* \| *precedence-criteria4*} \| **protocol** *protocol-name* \| **qos-group** *qos-group-value* \| **source-address** *mac address-destination*\| **vlan** {*vlan-id*\| *vlan-range* \| *vlan-combination*}}<br>**Example:**<br>`Router(config-if-spolicy-inline)# match any` | Specifies the classification criteria.<br>For more information and examples, see the *Cisco Media Monitoring Command Reference* . |
| **Step 6** | **flow monitor** {*monitor-name*\| **inline**}<br>**Example:**<br>`Router(config-if-spolicy-inline)# flow monitor inline` | Specifies an existing flow monitor to associate with a flow policy. If you do not want to use an existing flow monitor, you can use the **inline** option to configure a new one.<br>If needed, you can also use the **inline** option to specify a flow record and flow exporter. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 7** | **record** {*record-name*\| **default-rtp**\| **default-tcp**}<br>**Example:**<br>`Router(config-spolicy-inline-flowmon)# record default-tcp` | (Optional) If you do not want to use an existing flow monitor, and instead used the **inline** option, use this command to configure a flow record. |
| **Step 8** | **exporter** *exporter-name*<br>**Example:**<br>`Router(config-spolicy-inline-flowmon)# exporter exporter-4` | (Optional) If you do not want to use an existing flow monitor, and instead used the **inline** option, use this command to configure a flow exporter. |
| **Step 9** | exit<br>**Example:**<br>`Router(config-spolicy-inline-flowmon)# exit` | Returns to service-policy inline configuration mode. |
| **Step 10** | monitor metric ip-cbr<br>**Example:**<br>`Router(config-if-spolicy-inline)# monitor metric ip-cbr` | Enters IP-CBR monitor metric configuration mode. |
| **Step 11** | **rate layer3** {*byte-rate* {**bps** \| **kbps** \| **mbps** \| **gbps**} \| **packet**}<br>**Example:**<br>`Router(config-spolicy-inline-mipcbr)# rate layer3 248 mbps` | Specifies the rate for monitoring the metrics.<br>• *byte-rate* --Data rate in Bps, kBps, mBps, or gBps. The range is 1 to 65535.<br>• **packet** --Packet rate in packets per second. |
| **Step 12** | **exit**<br>**Example:**<br>`Router(config-spolicy-inline-mipcbr)# exit` | Returns to service-policy inline configuration mode. |
| **Step 13** | **monitor metric rtp**<br>**Example:**<br>`Router(config-if-spolicy-inline)# monitor metric rtp` | Enters RTP monitor metric configuration mode. |
| **Step 14** | **clock-rate** {*type-number*\| *type-name*} *rate*<br>**Example:**<br>`Router(config-spolicy-inline-mrtp)# clock-rate 8 9600` | Specifies the clock rate used to sample RTP video-monitoring metrics.<br>For more information about the clock-type numbers and names, see the *Cisco Media Monitoring Command Reference.*<br>The range for *rate* is 1 kHz to 192 kHz. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 15** | **max-dropout** *number*<br>**Example:**<br>`Router(config-spolicy-inline-mrtp)# max-dropout 2` | Specifies the maximum number of dropouts allowed when sampling RTP video-monitoring metrics. |
| **Step 16** | **max-reorder** *number*<br>**Example:**<br>`Router(config-spolicy-inline-mrtp)# max-reorder 4` | Specifies the maximum number of reorders allowed when sampling RTP video-monitoring metrics. |
| **Step 17** | **min-sequential** *number*<br>**Example:**<br>`Router(config-spolicy-inline-mrtp)# min-sequential 2` | Specifies the minimum number of sequental packets required to identify a stream as being an RTP flow. |
| **Step 18** | **ssrc maximum** *number*<br>**Example:**<br>`Router(config-spolicy-inline-mrtp)# ssrc maximum 20` | Specifies the maximum number of SSRCs that can be monitored within the same flow. A flow is defined by the protocol, source/destination address, and source/destination port). |
| **Step 19** | exit<br>**Example:**<br>`Router(config-spolicy-inline-mrtp)# exit` | Returns to service-policy inline configuration mode. |
| **Step 20** | **monitor parameters**<br>**Example:**<br>`Router(config-if-spolicy-inline)# monitor parameters` | Enters monitor parameters configuration mode. |
| **Step 21** | **flows** *number*<br>**Example:**<br>`Router(config-spolicy-inline-mparam)# flows 40` | Specifies the maximum number of flows for each monitor cache. |
| **Step 22** | **interval duration** *number*<br>**Example:**<br>`Router(config-spolicy-inline-mparam)# interval duration 40` | Specifies the duration of the intervals, in seconds, for collecting monitoring metrics. |
| **Step 23** | **history** *number*<br>**Example:** | Specifies the number of historical intervals of collected monitoring metrics to display. |

| | Command or Action | Purpose |
|---|---|---|
| | `Router(config-spolicy-inline-mparam)# history 4` | |
| **Step 24** | **timeout** *number*<br>**Example:**<br>`Router(config-spolicy-inline-mparam)# timeout 20` | Specifies the number of intervals before a stopped flow is removed from the database. |
| **Step 25** | **exit**<br>**Example:**<br>`Router(config-spolicy-inline-mparam)# exit` | Returns to service-policy inline configuration mode. |
| **Step 26** | **react** *ID* {**media-stop** \| **mrv** \| **rtp-jitter-average** \| **transport-packets-lost-rate**}<br>**Example:**<br>`Router(config-if-spolicy-inline)# react 6 rtp-jitter-average` | Enters a mode where you can specify what reaction occurs when a threshold is violated for the following metrics:<br>• *ID*-- ID for react configuration. Range is 1 to 65535.<br>• **media-stop** --No traffic is found for the flow.<br>• **mrv** --Ratio calculated by dividing the difference between the actual rate and the expected rate, by the expected rate.<br>• **rtp-jitter-average** --Average jitter.<br>• **transport-packets-lost-rate** --Ratio calculated by dividing the number of lost packets by the expected packet count. |
| **Step 27** | **action** {**snmp** \| **syslog**}<br>**Example:**<br>`Router(config-spolicy-inline-react)# action syslog` | Specifies how violations of the thresholds with be reported. |
| **Step 28** | **alarm severity** {**alert**\| **critical**\| **emergency**\| **error** \| **info**}<br>**Example:**<br>`Router(config-spolicy-inline-react)# alarm severity critical` | Specifies which level of alarm will be reported. |
| **Step 29** | **alarm type** {**discrete**\| **grouped**{**count** *number* \| **percent** *number*}}<br>**Example:**<br>`Router(config-pspolicy-inline-react)# alarm severity critical` | Specifies which types of levels are considered alarms that require reporting. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 30** | **threshold value** {**ge** *number* \| **gt** *number* \| **le** *number* \| **lt** *number* \| **range** *rng-start rng-end*}<br>**Example:**<br>`Router(config-spolicy-inline-react)# threshold value ge` | Specifies which types of levels values are considered alarms that require reporting. |
| **Step 31** | **end**<br>**Example:**<br>`Router(config-spolicy-inline-react)# end` | Exits the current configuration mode and returns to privileged EXEC mode. |

### What to do next

To check the configuration and status of your service policy, use the **show performance monitor status**command and **show performance monitor history**command.

# Verifying That Cisco Performance Monitor Is Collecting Data

To verify that Cisco Performance Monitor is collecting data, perform the following optional task.

**Note** Flows are correlated so that if the same policy is applied on the same input and output interface, the **show** command will display a single flow for the input and output interfaces and the interface name and direction for the flow are not displayed.

If no data is being collected, complete the remaining tasks in this section.

### Before you begin

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

where *filter* = {**ip** {*source-addr source-prefix* | **any**} {*dst-addr dst-prefix* | **any**} | {**tcp** | **udp**} {*source-addr source-prefix* | **any**} {[**eq**| **lt**| **gt** *number*| **range** *min max*| **ssrc** {*ssrc-number* | **any**} | {{*dst-addr dst-prefix* | **any**} **eq**| **lt**| **gt** *number*| **range** *min max*| **ssrc** {*ssrc-number* | **any**}}

## SUMMARY STEPS

1. **enable**
2. **show policy-map type performance-monitor** [**interface** *interface-name*][**class** *class-name*][**input** | **output**]
3. **show performance monitor status** [**interface** *interface name*[*filter*] | **policy** *policy-map-name* **class** *class-map-name*[*filter*]} | *filter*]
4. **show performance monitor history** [**interval**{**all**| *number*[**start** *number*]} | **interface** *interface name*[*filter*] | **policy** *policy-map-name* **class** *class-map-name*[*filter*]} | *filter* ]

## DETAILED STEPS

**Step 1** **enable**

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

**Example:**

```
Device> enable
Device#
```

**Step 2** **show policy-map type performance-monitor** [**interface** *interface-name*][**class** *class-name*][**input** | **output**]

For a description of the fields displayed by this command, see *Cisco Media Monitoring Command Reference*.

The following example shows the output for one flow policy:

**Example:**

```
Policy Map type performance-monitor PM-POLICY-4
  Class PM-CLASS-4
    flow monitor PM-MONITOR-4
      record PM-RECORD-4
      exporter PM-EXPORTER-4
    monitor parameters
      interval duration 30
      timeout 10
      history 10
      flows 8000
    monitor metric rtp
      min-sequential 5
      max-dropout 5
      max-reorder 5
      clock-rate default 90000
      ssrc maximum 5
```

***Table 4: show policy-map type performance-monitor Field Descriptions***

| Field | Description |
|---|---|
| Policy Map type performance-monitor | Name of the Cisco Performance Monitor flow policy. |
| flow monitor | Name of the Cisco Performance Monitor flow monitor. |
| record | Name of the Cisco Performance Monitor flow record. |
| exporter | Name of the Cisco Performance Monitor flow exporter. |
| monitor parameter | Parameters for the flow policy. |
| interval duration | The configured duration of the collection interval for the policy. |
| timeout | The configured amount of time wait for a response when collecting data for the policy. |
| history | The configured number of historical collections to keep for the policy. |

| Field | Description |
|---|---|
| flows | The configured number of flows to collect for the policy. |
| monitor metric rtp | RTP metrics for the flow policy. |
| min-sequential | The configured minimum number of packets in a sequence used to classify an RTP flow. |
| max-dropout | The configured maximum number of packets to ignore ahead of the current packet in terms of sequence number. |
| max-reorder | The configured maximum number of packets to ignore behind the current packet in terms of sequence number. |
| clock-rate default | The configured clock rate for the RTP packet timestamp clock that is used to calculate the packet arrival latency. |
| ssrc maximum | The configured maximum number of SSRCs that can be monitored within the same flow. A flow is defined by the protocol, source/destination address, and source/destination port. The range is from 1 to 50. |

**Step 3** **show performance monitor status** [**interface** *interface name*[*filter*] | **policy** *policy-map-name* **class** *class-map-name*[*filter*]} | *filter*]

where *filter* = {**ip** {*source-addr source-prefix* | **any**} {*dst-addr dst-prefix* | **any**} | {**tcp** | **udp**} {*source-addr source-prefix* | **any**} {[**eq**| **lt**| **gt** *number*| **range** *min max*| **ssrc** {*ssrc-number* | **any**} | {{*dst-addr dst-prefix* | **any**} **eq**| **lt**| **gt** *number*| **range** *min max*| **ssrc** {*ssrc-number* | **any**}}

This command displays the cumulative statistics for the specified number of most recent intervals. The number of intervals is configured using the **history** command. The default settings for this commands is 10 of the most recent collection intervals. The duration of collection intervals is specified by the **interval duration** command.

To view statistics for other intervals, use the **show performance monitor history** command as described in the next step. For more information about these commands, see the *Cisco Media Monitoring Command Reference*

**Step 4** **show performance monitor history** [**interval**{**all**| *number*[**start** *number*]} | **interface** *interface name*[*filter*] | **policy** *policy-map-name* **class** *class-map-name*[*filter*]} | *filter* ]

where *filter* = {**ip** {*source-addr source-prefix* | **any**} {*dst-addr dst-prefix* | **any**} | {**tcp** | **udp**} {*source-addr source-prefix* | **any**} {[**eq**| **lt**| **gt** *number*| **range** *min max*| **ssrc** {*ssrc-number* | **any**} | {{*dst-addr dst-prefix* | **any**} **eq**| **lt**| **gt** *number*| **range** *min max*| **ssrc** {*ssrc-number* | **any**}}

This command displays the statistics collected by Cisco Performance Monitor during any or all intervals, including the current one. The duration of collection intervals is specified by the **interval duration** command.

For more information about this command, see the *Cisco Media Monitoring Command Reference.*

The following example shows the output for the **show performance monitor history** command**:**

**Note** If the same policy is applied on the same input and output interface, the display shows a single flow for the input and output interfaces and the interface name and direction for the flow are not displayed.

**Example:**

```
Codes: *   - field is not configurable under flow record
       NA  - field is not applicable for configured parameters
```

```
Match: ipv4 source address = 21.21.21.1, ipv4 destination address = 1.1.1.1,
transport source-port = 10240, transport destination-port = 80, ip protocol = 6,
 Policy: RTP_POL, Class: RTP_CLASS

 start time                                14:57:34
                                           ============
 *history bucket number                   : 1
 routing forwarding-status                : Unknown
 transport packets expected counter       : NA
 transport packets lost counter           : NA
 transport round-trip-time          (msec) : 4
 transport round-trip-time sum      (msec) : 8
 transport round-trip-time samples        : 2
 transport event packet-loss counter      : 0
 interface input                          : Null
 interface output                         : Null
 counter bytes                            : 8490
 counter packets                          : 180
 counter bytes rate                       : 94
 counter client bytes                     : 80
 counter server bytes                     : 200
 counter client packets                   : 6
 counter server packets                   : 6
 transport tcp window-size minimum        : 1000
 transport tcp window-size maximum        : 2000
 transport tcp window-size average        : 1500
 transport tcp maximum-segment-size       : 0
 application media bytes counter          : 1270
 application media bytes rate             : 14
 application media packets counter        : 180
 application media event                  : Stop
 monitor event                            : false

 [data set,id=257] Global session ID|Multi-party session ID|
 [data] 11                      |22
```

***Table 5: show performance monitor status and show performance-monitor history Field Descriptions***

| Field | Description |
|---|---|
| history bucket number | Number of the bucket of historical data collected. |

| Field | Description |
|---|---|
| routing forwarding-status reason | |

| Field | Description |
|---|---|
| | Forwarding status is encoded using eight bits with the two most significant bits giving the status and the six remaining bits giving the reason code.<br>Status is either unknown (00), Forwarded (10), Dropped (10) or Consumed (11).<br>The following list shows the forwarding status values for each status category.<br>**Unknown**<br>• 0<br>**Forwarded**<br>• Unknown 64<br>• Forwarded Fragmented 65<br>• Forwarded not Fragmented 66<br>**Dropped**<br>• Unknown 128,<br>• Drop ACL Deny 129,<br>• Drop ACL drop 130,<br>• Drop Unroutable 131,<br>• Drop Adjacency 132,<br>• Drop Fragmentation & DF set 133,<br>• Drop Bad header checksum 134,<br>• Drop Bad total Length 135,<br>• Drop Bad Header Length 136,<br>• Drop bad TTL 137,<br>• Drop Policer 138,<br>• Drop WRED 139,<br>• Drop RPF 140,<br>• Drop For us 141,<br>• Drop Bad output interface 142,<br>• Drop Hardware 143,<br>**Consumed**<br>• Unknown 192,<br>• Terminate Punt Adjacency 193,<br>• Terminate Incomplete Adjacency 194, |

| Field | Description |
|---|---|
| | • Terminate For us 195 |
| transport packets expected counter | Number of packets expected. |
| transport packets lost counter | Number of packets lost. |
| transport round-trip-time (msec) | Number of milliseconds required to complete a round trip. |
| transport round-trip-time sum (msec) | Total number of milliseconds required to complete a round trip for all samples. |
| transport round-trip-time samples | Total number of samples used to calculate a round trip times |
| transport event packet-loss counter | Number of loss events (number of contiguous sets of lost packets). |
| interface input | Incoming interface index. |
| interface output | Outgoing interface index. |
| counter bytes | Total number of bytes collected for all flows. |
| counter packets | Total number of IP packets sent for all flows. |
| counter bytes rate | Average number of packets or bits (as configured) processed by the monitoring system per second during the monitoring interval for all flows. |
| counter client bytes | Number of bytes sent by the client. |
| counter server bytes | Number of bytes sent by the server. |
| counter client packets | Number of packets sent by the client. |
| counter servers packets | Number of packets sent by the server. |
| transport tcp window-size-maximum | Maximum size of the TCP window. |
| transport tcp window-size-minimum | Minimum size of the TCP window. |
| transport tcp window-size-average | Average size of the TCP window. |
| transport tcp maximum-segment-size | Maximum TCP segment size. |
| application media bytes counter | Number of IP bytes from by media applications received for a specific media stream. |
| application media bytes rate | Average media bit rate (bps) for all flows during the monitoring interval. |
| application media packets counter | Number of IP packets produced from media applications received for a specific media stream. |
| application media event | Bit 1 is not used. Bit 2 indicates that no media application packets were seen, in other words, a Media Stop Event occurred. |

| Field | Description |
|---|---|
| monitor event | Bit 1 indicates that one of the thresholds specified by a react statement for the flow was crossed at least once in the monitoring interval. Bit 2 indicates that there was a loss-of-confidence in measurement. |

# Displaying the Performance Monitor Cache and Clients

To display the cache and the clients for Cisco Performance Monitor, perform the following optional task.

## SUMMARY STEPS

1. **enable**
2. **show performance monitor cache** [**policy** *policy-map-name* **class** *class-map-name*][**interface** *interface name*]
3. **show performance monitor clients detail all**

## DETAILED STEPS

### Step 1 enable

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

**Example:**

```
Device> enable
Device#
```

### Step 2 show performance monitor cache [policy *policy-map-name* class *class-map-name*][interface *interface name*]

**Example:**

```
MMON Metering Layer Stats:
  static pkt cnt: 3049
  static cce sb cnt: 57
  dynamic pkt cnt: 0
  Cache type:                              Permanent
  Cache size:                                   2000
  Current entries:                                 8
  High Watermark:                                  9
  Flows added:                                     9
  Updates sent             (  1800 secs)           0
IPV4 SRC ADDR    IPV4 DST ADDR    IP PROT  TRNS SRC PORT  TRNS DST PORT
ipv4 ttl ipv4 ttl min ipv4 ttl max  ipv4 dscp bytes long perm pktslong perm  user space vm
==========================================================================================
10.1.1.1         10.1.2.3              17           4000           1967
0            0             0  0x00                        80
1 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
```

```
0x00000000
10.1.1.1         10.1.2.3              17           6000           1967
0             0             0  0x00                          80
1  0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000
10.1.1.1         10.1.2.3              17           4000           2000
0             0             0  0x00                          44
1  0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000
10.1.1.1         10.1.2.3               6           6000           3000
0             0             0  0x00                          84
2  0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000
10.1.1.1         10.1.2.3              17           1967           6001
0             0             0  0x00                          36
1  0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000
10.1.1.1         10.1.2.3              17           1967           4001
0             0             0  0x00                          36
1  0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000
10.1.1.1         10.1.2.3               6           3001           6001
0             0             0  0x00                         124
3  0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000
10.1.1.1         10.1.2.3              17           2001           4001
0             0             0  0x00                          44
1  0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000
```

**Step 3** **show performance monitor clients detail all**

**Example:**

```
Client name for ID 1 : Mediatrace-131419052
   Type: Mediatrace
   Age: 443 seconds
   Monitor Object: _MMON_DYN_-class-map-69
        Flow spec: (dvmc-acl#47) 10.10.130.2 1000 10.10.132.2 2000 17
        monitor parameters
                interval duration 60
                timeout 2
                history 1
                flows 100
        monitor metric rtp
                min-sequential 10
                max-dropout 5
                max-reorder 5
                clock-rate 112 90000
                clock-rate default 90000
                ssrc maximum 20
        monitor metric ip-cbr
                rate layer3 packet 20
        Flow record: dvmc_fnf_fdef_47
                Key fields:
                        ipv4 source address
                        ipv4 destination address
                        transport source-port
                        transport destination-port
                        ip protocol
                Non-key fields:
                        monitor event
                        application media event
                        routing forwarding-status
                        ip dscp
                        ip ttl
                        counter bytes rate
                        application media bytes rate
                        transport rtp jitter mean
                        transport packets lost counter
                        transport packets expected counter
                        transport event packet-loss counter
                        transport packets lost rate
                        timestamp interval
                        counter packets dropped
                        counter bytes
                        counter packets
                        application media bytes counter
                        application media packets counter
        Monitor point: _MMON_DYN_-policy-map-70 GigabitEthernet0/3 output
        Classification Statistic:
                matched packet: 545790
                matched byte: 64403220
```

# Displaying the Clock Rate for Cisco Performance Monitor Classes

To display the clock rate for one or more classes, perform the following optional task.

## SUMMARY STEPS

1. **enable**
2. **show performance monitor clock rate** [**policy** *policy-map-name* **class** *class-map-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 performance monitor clock rate** [**policy** *policy-map-name* **class** *class-map-name*]

If no class name is specified, information for all classes are displayed.

**Example:**

```
Device# show performance monitor clock rate policy all-apps class telepresence-CS4
Load for five secs: 6%/2%; one minute: 5%; five minutes: 5% Time source is NTP, 17:41:35.508 EST Wed
 Feb 16 2011
RTP clock rate for Policy: all-apps, Class: telepresence-CS4
     Payload type     Clock rate(Hz)
     pcmu    (0  )     8000
     gsm     (3  )     8000
     g723    (4  )     8000
     dvi4    (5  )     8000
     dvi4-2  (6  )     16000
     lpc     (7  )     8000
     pcma    (8  )     8000
     g722    (9  )     8000
     l16-2   (10 )     44100
     l16     (11 )     44100
     qcelp   (12 )     8000
     cn      (13 )     8000
     mpa     (14 )     90000
     g728    (15 )     8000
     dvi4-3  (16 )     11025
     dvi4-4  (17 )     22050
     g729    (18 )     8000
     celb    (25 )     90000
     jpeg    (26 )     90000
     nv      (28 )     90000
     h261    (31 )     90000
     mpv     (32 )     90000
     mp2t    (33 )     90000
     h263    (34 )     90000
             (96 )     48000
             (112)     90000
     default           90000
```

# Displaying the Current Status of a Flow Monitor

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

### Before you begin

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

### SUMMARY STEPS

1. **enable**
2. **show flow monitor type performance-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 flow monitor type performance-monitor**

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

**Example:**

```
Device# show flow monitor type performance-monitor
Flow Monitor type performance-monitor monitor-4:
  Description:           User defined
  Flow Record:           record-4
  Flow Exporter:         exporter-4
  No. of Inactive Users: 0
  No. of Active Users:   0
```

# Verifying the Flow Monitor Configuration

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

### Before you begin

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

### 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
Current configuration:
!
flow monitor FLOW-MONITOR-1
 description Used for basic IPv4 traffic analysis
 record netflow ipv4 original-input
!
!
flow monitor FLOW-MONITOR-2
 description Used for basic IPv6 traffic analysis
 record netflow ipv6 original-input
!
```

# Verifying That Cisco IOS Flexible NetFlow and Cisco Performance Monitor Is Enabled on an Interface

To verify that Flexible NetFlow and Cisco Performance Monitor is enabled on an interface, perform the following optional task.

**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:**

```
Router> enable
Router#
```

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

The **show flow interface** command verifies that Flexible NetFlow and Cisco Performance Monitor is enabled on an interface.

**Example:**

```
Router# show flow interface ethernet 0/0
Interface Ethernet0/0
  FNF:  monitor:         FLOW-MONITOR-1
        direction:       Input
        traffic(ip):     on
  FNF:  monitor:         FLOW-MONITOR-2
        direction:       Input
        traffic(ipv6):   on
```

# Displaying the Flow Monitor Cache

To display the data in the flow monitor cache, perform the following optional task.

### Before you begin

The interface to which you applied the input flow monitor must be receiving traffic that meets the criteria defined by the original flow record before you can display the flow data 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 the flow data in the cache for a flow monitor.

**Example:**

```
Device# show flow monitor name FLOW-MONITOR-1 cache format record
Cache type:                           Normal
```

```
  Cache size:                             4096
  Current entries:                           8
  High Watermark:                            8
  Flows added:                              24
  Flows aged:                               16
    - Active timeout   (  1800 secs)         0
    - Inactive timeout (    15 secs)        16
    - Event aged                             0
    - Watermark aged                         0
    - Emergency aged                         0
IPV4 SOURCE ADDRESS:       10.251.10.1
IPV4 DESTINATION ADDRESS:  172.16.10.2
TRNS SOURCE PORT:          0
TRNS DESTINATION PORT:     2048
INTERFACE INPUT:           Et0/0
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
counter bytes:             733500
counter packets:           489
timestamp first:           720892
timestamp last:            975032
.
.
.
IPV4 SOURCE ADDRESS:       172.16.6.1
IPV4 DESTINATION ADDRESS:  224.0.0.9
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:           973804
timestamp last:            973804
Device# show flow monitor name FLOW-MONITOR-2 cache format record
Cache type:                           Normal
  Cache size:                             4096
  Current entries:                           6
  High Watermark:                            8
  Flows added:                            1048
  Flows aged:                             1042
    - Active timeout   (  1800 secs)        11
    - Inactive timeout (    15 secs)      1031
    - Event aged                             0
    - Watermark aged                         0
    - Emergency aged                         0
IPV6 FLOW LABEL:           0
```

```
IPV6 EXTENSION MAP:         0x00000040
IPV6 SOURCE ADDRESS:        2001:DB8:1:ABCD::1
IPV6 DESTINATION ADDRESS:   2001:DB8:4:ABCD::2
TRNS SOURCE PORT:           3000
TRNS DESTINATION PORT:      55
INTERFACE INPUT:            Et0/0
FLOW DIRECTION:             Input
FLOW SAMPLER ID:            0
IP PROTOCOL:                17
IP TOS:                     0x00
ip source as:               0
ip destination as:          0
ipv6 next hop address:      ::
ipv6 source mask:           /48
ipv6 destination mask:      /0
tcp flags:                  0x00
interface output:           Null
counter bytes:              521192
counter packets:            9307
timestamp first:            9899684
timestamp last:             11660744
.
.
.
IPV6 FLOW LABEL:            0
IPV6 EXTENSION MAP:         0x00000000
IPV6 SOURCE ADDRESS:        FE80::A8AA:BBFF:FEBB:CC03
IPV6 DESTINATION ADDRESS:   FF02::9
TRNS SOURCE PORT:           521
TRNS DESTINATION PORT:      521
INTERFACE INPUT:            Et0/0
FLOW DIRECTION:             Input
FLOW SAMPLER ID:            0
IP PROTOCOL:                17
IP TOS:                     0xE0
ip source as:               0
ip destination as:          0
ipv6 next hop address:      ::
ipv6 source mask:           /10
ipv6 destination mask:      /0
tcp flags:                  0x00
interface output:           Null
counter bytes:              92
counter packets:            1
timestamp first:            11653832
timestamp last:             11653832
```

# 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** [*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 flow exporter** [*exporter-name*]

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 Chicago datacenter
  Transport Configuration:
    Destination IP address: 172.16.10.2
    Source IP address:      172.16.7.1
    Transport Protocol:     UDP
    Destination Port:       65
    Source Port:            56041
    DSCP:                   0x0
    TTL:                    255
```

# Verifying the Flow Exporter Configuration

To verify the configuration commands that you entered to configure the flow exporter, 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...
!
flow exporter EXPORTER-1
 description Exports to datacenter
 destination 172.16.10.2
 transport udp 65
!
```

# Enabling Debugging

To enable debugging for Cisco Performance Monitor, perform the following optional task in privileged EXEC mode.

### SUMMARY STEPS

1. **debug performance monitor** {**database** | **dynamic** | **event** | **export** | **flow-monitor** | **metering** | **provision** | **sibling** | **snmp** | **tca** | **timer**}

### DETAILED STEPS

**debug performance monitor** {**database** | **dynamic** | **event** | **export** | **flow-monitor** | **metering** | **provision** | **sibling** | **snmp** | **tca** | **timer**}

The **debug performance monitor** command enables debugging for the following performance monitor components:

- Flow database
- Dynamic monitoring
- Performance events
- Exporting
- Flow monitors
- Metering layer
- Provisioning
- Sibling management
- SNMP
- TCA
- Timers

The following example shows how to enable debugging for dynamic monitoring:

**Example:**

```
Device# debug performance monitor dynamic
```

# Configuration Example for Cisco Performance Monitor

## Example Monitor for Lost RTP Packets and RTP Jitter

This example show a configuration that monitors the number of lost RTP packets, the amount of RTP jitter, and other basic statistics for the **gig1** interface. In this example, Cisco Performance Monitor is also configured to make an entry in the syslog when the any of the following events occur on the interface:

- The percentage of lost RTP packets is between 5 percent and 9 percent.
- The percentage of lost RTP packets is greater than 10 percent.
- A media stop event has occurred.

```
! Set the filter spec for the flows to monitor.
access-list 101 ip permit host 10.10.2.20 any
! Use the flow record to define the flow keys and metric to collect.
flow record type performance-monitor video-monitor-record
 match ipv4 source
 match ipv4 destination
 match transport source-port
 match transport destination-port
 match rtp ssrc
 collect timestamp
 collect counter byte
 collect counter packet
 collect mse
 collect media-error
 collect counter rtp interval-jitter
 collect counter rtp packet lost
 collect counter rtp lost event
! Set the exporting server. The export message format is based on FNFv.9.
flow export video-nms-server
 export-protocol netflow-v9
 destination cisco-video-management
 transport udp 32001
! Set the flow filter in the class-map.
class-map match-all video-class
 access-group  ipv4 101
! Set the policy map with the type performance-monitor for video monitor.
policy-map type performance-monitor video-monitor
 ! Set the video monitor actions.
 class video-class
  ! Specify where the metric data is being exported to.
  export  flow video-nms-server
  flow monitor inline
   record video-monitor-record
! Set the monitoring modeling parameters.
monitor parameters
 ! Set the measurement timeout to 10 secs.
 interval duration 10
```

```
 ! Set the timeout to 10 minutes.
 timeout 10
 ! Specify that 30 flow intervals can be kept in performance database.
 history 30
 priority 7
 ! Set rtp flow verification criteria.
 monitor metric rtp
 ! Configure a RTP flow criteria: at least 10 packets in sequence.
 min-sequential   10
 ! Ignore packets that are more than 5 packet ahead in terms of seq  number.  max-dropout
 5
 ! Ignore packets that are more than 5 packets behind in terms of seq  number.
 max-reorder 5
 ! Set the clock rate frequency for rtp packet timestamp clock.
 clock-rate 89000
 ! Set the maximum number of ssrc allowed within this class.
 ssrc maximum  100
 ! Set TCA for alarm.
 react 100 transport-packets-lost-rate
  description critical  TCA
  ! Set the threshold to greater than 10%.
  threshold gt 10
  ! Set the threshold to the average number based on the last five intervals.
  threshold type average 5
  action  syslog
  alarm severity critical
 react 110 transport-packets-lost-rate
  description medium TCA
  ! Set the threshold to between 5% and 9% of packet lost.
  threshold range gt 5 le 9
  threshold type average 10
  action  syslog
  alarm type grouped percent 30
 react 3000 media-stop
  action syslog
  alarm severity critical
  alarm type grouped percent 30

interface gig1
 service-policy type performance-monitor video-mon in
```

# Where to Go Next

For more information about configuring the products in the Medianet product family, see the other chapter in this guide or see the *Cisco Media Monitoring Configuration Guide*.

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| Design, configuration, and troubleshooting resources for Performance Monitor and other Cisco Medianet products, including a Quick Start Guide and Deployment Guide. | See the Cisco Medianet Knowledge Base Portal, located at http://www.cisco.com/web/solutions/medianet/knowledgebase/index.html |
| IP addressing commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples | *Cisco Media Monitoring Command Reference* |
| Cisco IOS commands | Cisco IOS Master Commands List, All Releases |
| Configuration commands for Flexible NetFlow | *Cisco IOS Flexible NetFlow Command Reference* |
| Overview of Flexible NetFlow | "Cisco IOS Flexible NetFlow Overview" |
| Flexible NetFlow Feature Roadmap | "Cisco IOS Flexible NetFlow Features Roadmap" |
| Configuring flow exporters to export Flexible NetFlow data. | "Configuring Data Export for Cisco IOS Flexible NetFlow with Flow Exporters" |
| Customizing Flexible NetFlow | "Customizing Cisco IOS Flexible NetFlow Flow Records and Flow Monitors" |
| Configuring flow sampling to reduce the overhead of monitoring traffic with Flexible NetFlow | "Using Cisco IOS Flexible NetFlow Flow Sampling to Reduce the CPU Overhead of Analyzing Traffic" |
| Configuring Flexible NetFlow using predefined records | "Configuring Cisco IOS Flexible NetFlow with Predefined Records" |
| Using Flexible NetFlow Top N Talkers to analyze network traffic | "Using Cisco IOS Flexible NetFlow Top N Talkers to Analyze Network Traffic" |
| Configuring IPv4 multicast statistics support for Flexible NetFlow | "Configuring IPv4 Multicast Statistics Support for Cisco IOS Flexible NetFlow" |

### Standards

| Standard | Title |
|---|---|
| None | — |

### MIBs

| MIB | MIBs Link |
|---|---|
| • CISCO-FLOW-MONITOR-TC-MIB<br>• CISCO-FLOW-MONITOR-MIB<br>• CISCO-RTP-METRICS-MIB<br>• CISCO-IP-CBR-METRICS-MIB | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:<br>http://www.cisco.com/go/mibs |

### RFCs

| RFC | Title |
|---|---|
| RFC 3954 | *Cisco Systems NetFlow Services Export Version 9*<br>http://www.ietf.org/rfc/rfc3954.txt |
| RFC 3550 | *RTP: A Transport Protocol for Real-Time Applications*<br>http://www.ietf.org/rfc/rfc3550.txt |

### 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 Cisco Performance Monitor

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.

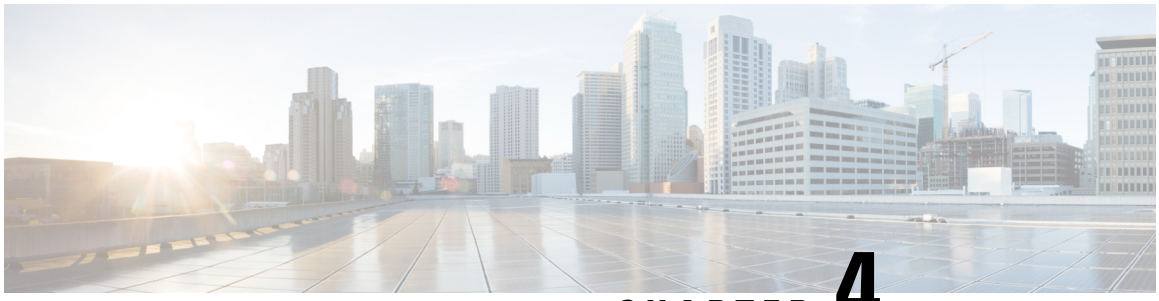
Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

*Table 6: Feature Information for Cisco Performance Monitor*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Cisco Performance Monitor 1.0 | 15.1(3)T<br>12.2(58)SE<br>15.1(4)M1<br>15.0(1)SY<br>Cisco IOS XE Release 3.5S<br>15.1(1)SG<br>Cisco IOS XE Release 3.3 SG<br>15.1(2)SY | This feature enables you to monitor the flow of packets in your network and become aware of any issues that might impact the flow before it starts to significantly impact your applications' performance.<br>Support for this feature was added for Cisco ASR 1000 Series Aggregation Services routers in Cisco IOS XE Release 3.5S.<br>There are some limitations to the monitoring of ingress or egress data on certain types of interfaces for the Cisco IOS XE Release 3.3 SG and Cisco IOS release 15.1(1)SG. For more information, see the "Limitations" section.<br>For all other releases, the following commands were introduced or modified by this feature: **action**(policy react and policy inline react), **alarm severity** (policy react and policy inline react), **alarm type**(policy react and policy inline react), **class-map**, **clock-rate**(policy RTP), **collect application media**, **clear fm performance-monitor counters**, **collect counter**, **collect flow direction**, **collect interface**, **collect ipv4**, **collect ipv4 destination**, **collect ipv4 source**, **collect ipv4 ttl**, **collect monitor event**, **collect routing**, **collect timestamp interval**, **collect transport event packet-loss counter**, **collect transport packets**, **collect transport rtp jitter**, **debug fm performance-monitor counters**, **debug performance-monitor counters**, **description** (Performance Monitor), **destination dscp** (Flexible NetFlow), **export-protocol**, **exporter**, **flow monitor type performance-monitor**, **flow record type performance-monitor**, **flows**, **history** (monitor parameters), **interval duration**, **match access-group**, **match any**, **match class-map**, **match cos**, **match destination-address mac**, **match discard-class**, **match dscp**, **match flow**, **match fr-de**, **match fr-dlci**, **match input-interface**, **match ip dscp**, **match ip precedence**, **match ip rtp**, **match ipv4**, **match ipv4 destination**, **match ipv4 source**, **match mpls experimental topmost**, **match not**, **match packet length** (class-map), **match precedence**, **match protocol**, **match qos-group**, **match source-address mac**, **match transport destination-port**, **match transport rtp ssrc**, **match transport source-port**, **match vlan**, **max-dropout** (policy RTP), **max-reorder** (policy RTP), **min-sequential** (policy RTP), **monitor metric ip-cbr**, **monitor metric rtp**, **monitor parameters**, **option** (Flexible NetFlow), **output-features**, **platform performance-monitor rate-limit**, **policy-map type performance-monitor**, **rate layer3**, **react** (policy), **record** (Performance Monitor), **rename** (policy), **service-policy type performance-monitor**, **show performance monitor history**, **show performance monitor status**, **show platform hardware acl entry interface**, **show platform software ccm**, **show platform software feature-manager performance-monitor**, **show platform software feature-manager tcam**, **show policy-map type performance-monitor**, **snmp-server host**, **snmp-server enable traps flowmon**, **snmp mib flowmon alarm history** , **source**(Flexible NetFlow), **ssrc maximum**, **template data timeout**, **threshold value** (policy react and policy inline react), **timeout** (monitor parameters), **transport** (Flexible NetFlow), and **ttl** (Flexible NetFlow). |

| Feature Name | Releases | Feature Information |
|---|---|---|
| Cisco Performance Monitor (phase 2) | 15.2(2)T<br>Cisco IOS XE Release 3.5S | This feature enables you monitor IPv6 fields and also use all other Flexible Netflow collect and match commands not supported in the previous release.<br>Flows are now correlated so that if the same policy is applied on the same input and output interface, the show command will display a single flow for the input and output interfaces.<br>Support for this feature was added for Cisco ASR 1000 Series Aggregation Services routers in Cisco IOS XE Release 3.5S.<br>The following commands were introduced or modified by this feature: **collect datalink mac**, **collect ipv4 fragmentation**, **collect ipv4 section**, **collect ipv4 total-length**, **collect ipv6**, **collect ipv6 destination**, **collect ipv6 extensionmap**, **collect ipv6 fragmentation**, **collect ipv6 hop-count**, **collect ipv6 length**, **collect ipv6 section**, **collect ipv6 source**, **collect routing is-multicast**, **collect routing multicast replication-factor**, **collect timestamp sys-uptime**, **collect transport**, **collect transport icmp ipv4**, **collect transport icmp ipv6**, **collect transport tcp**, **collect transport udp**, **match application name**, **match connection transaction-id**, **match datalink dot1q vlan**, **match datalink mac**, **match datalink vlan**, **match interface**, **match ipv4 fragmentation**, **match ipv4 section**, **match ipv4 total-length**, **match ipv4 ttl**, **match ipv6**, **match ipv6 destination**, **match ipv6 extension map**, **match ipv6 fragmentation**, **match ipv6 hop-limit**, **match ipv6 length**, **match ipv6 section**, **match ipv6 source**, **match routing**, **match routing is-multicast**, **match routing multicast replication-factor**, **match transport**, **match transport icmp ipv4**, **match transport icmp ipv6**, **match transport tcp**, **match transport udp** |
| Cisco Performance Monitor (phase 3) | 15.2(3)T | This feature enables you to configure multiple exporters and monitor metadata fields and new TCP metrics.<br>The following commands were introduced or modified by this feature: **collect application**, **collect transport tcp bytes out-of-order**, **collect transport packets out-of-order**, **collect transport tcp maximum-segment-size**, **collect transport tcp window-size maximum**, **collect transport tcp window-size minimum**, **collect transport tcp window-size average**, **match application**, **match transport tcp bytes out-of-order**, **match transport packets out-of-order**, **match transport tcp maximum-segment-size**, **match transport tcp window-size maximum**, **match transport tcp window-size minimum**, **match transport tcp window-size average** |
| Performance Monitoring - IPv6 support | Cisco IOS XE Release 3.6S | This feature enables you to attach a monitor to IPv6 interfaces.<br>Support for this feature was added for Cisco ASR 1000 Series Aggregation Services routers in Cisco IOS XE Release 3.6S. |

| Feature Name | Releases | Feature Information |
|---|---|---|
| Performance Monitoring - transport packet out of order | Cisco IOS XE Release 3.6S | This feature enables you to monitor the total number of out-of-order TCP packets.<br>Support for this feature was added for Cisco ASR 1000 Series Aggregation Services routers in Cisco IOS XE Release 3.6S.<br>The following commands were introduced or modified by this feature: **collect transport tcp bytes out-of-order** and **collect transport packets out-of-order**. |
| Flexible NetFlow: IPFIX Export Format | 15.2(4)M | Enables sending export packets using the IPFIX export protocol. The export of extracted fields from NBAR is only supported over IPFIX.<br>The following command was introduced: **export-protocol**. |



CHAPTER 4

# Metrics for Assurance Monitoring

Metrics for Assurance monitoring refers to Assurance-related metrics collected per network application, for flows forwarded through specific interfaces, to support Assurance monitoring by Cisco DNA Center. FNF provides a pair of record types (for IPv4 and IPv6) to collect this data. Monitoring for Assurance is optimized to provide better than typical performance for FNF monitors.



# Feature Information for Metrics for Assurance Monitoring

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 7: Feature Information for Metrics for Assurance Monitoring***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Metrics for Assurance Monitoring | Cisco IOS XE Gibraltar 16.10.1 | FNF provides a pair of record types to collect data for Assurance, optimized to provide better than typical performance for FNF monitors. |

# Information About Metrics for Assurance Monitoring

## Overview

### DNA Center Assurance

Cisco DNA Center Assurance collects and analyzes network data to help provide better and more consistent network performance. DNA Center uses Flexible NetFlow (FNF) to collect specific network metrics for Assurance, providing quantitative and qualitative information about devices in the network. The FNF records designed for Assurance-related metrics are specially optimized for improved performance.

FNF provides a pair of record types (for IPv4 and IPv6) to collect data for Assurance. Monitoring Assurance metrics using these dedicated record types is optimized to provide better performance, as compared with typical FNF monitors configured to collect the same metrics. (Modifying the records cancels the dedicated performance enhancements for Assurance, and may prevent attaching a monitor to an interface.)

### Manual Configuration

In typical use, DNA Center configures the monitors to collect data for Assurance, without requiring user input. However, it is also possible to use these record types manually.

## Metrics Collected for Assurance

Most of the metrics collected for Assurance are metrics that have been available through FNF and other monitor types, but when they are collected specifically for Assurance records, some metrics may behave slightly differently.

***Table 8: Metrics***

| Metric | Information |
|---|---|
| match ipv4/ipv6 version | IPv4/IPv6 version from IPv4/IPv6 header.<br>[1] |
| match ipv4/ipv6 protocol | Layer4 protocol from the IPv4/IPv6 header. |
| match application name | Application ID |
| match connection client ipv4/ipv6 address | Field name: clientIPv4/IPv6Address<br>IPv4/IPv6 client address in the IP packet header. The client is the device that triggered the session creation, and remains the same for the life of the session.<br>[2] |

| Metric | Information |
|---|---|
| match connection server ipv4/ipv6 address | Field name: serverIPv4/IPv6Address<br>IPv4/IPv6 server address in the IP packer header. The server is the device that replies to the client, and remains the same for the life of the session.<br>[2] |
| match connection server transport port | Field name: serverTransportPort<br>Server transport port identifier. This may be the source or destination transport port. The server is the device that replies to the client, and remains the same for the life of the session.<br>[2] |
| match flow observation point | Field name: observationPointId<br>Identifier of an observation point unique for each observation domain.<br>[2] |
| collect connection initiator | Field name: biflowDirection<br>Description of the direction assignment method used to assign the Biflow Source and Destination.<br>[2] |
| collect flow direction | Direction (ingress/egress) of the flow observed at the observation point. |
| collect routing vrf input | Field name: ingressVRFID<br>(Applies only to routers, not wireless controllers)<br>VRF ID from incoming packets on a router. If a packet arrives on an interface that does not belong to a VRF, a VRF ID of 0 is recorded. |
| collect wireless client mac address | (Applies only to wireless controllers)<br>Field name: staMacAddress<br>The IEEE 802 MAC address of a wireless station (STA). |
| collect timestamp absolute first | Field name: flowStartMilliseconds<br>The absolute timestamp of the first packet of the flow. |
| collect timestamp absolute last | Field name: flowEndMilliseconds<br>The absolute timestamp of the last packet of the flow. |
| collect connection new-connections | Field name: connectionCountNew<br>This information element counts the number of TCP or UDP connections which were opened during the observation period. The observation period may be specified by the flow start and end timestamps.<br>[2] |

| Metric | Information |
|---|---|
| collect connection server counter packets long | Field name: serverPackets<br>Number of layer 4 packets in a flow from the server. The server is the device that replies to the client, and remains the same for the life of the session.<br>[2] |
| collect connection server counter bytes network long | Field name: serverOctets<br>Overall IP packet bytes in a flow from the server. The server is the device that replies to the client, and remains the same for the life of the session.<br>[2] |
| collect connection client counter packets long | Field name: clientPackets<br>Number of layer 4 packets in a flow from the client. The client is the device that triggered the session creation, and remains the same for the life of the session.<br>[2] |
| collect connection client counter bytes network long | Overall IP packet bytes from client to server.<br>[2] |
| collect connection delay network client-to-server sum | Field name: sumNwkTime<br>Network delay is the round-trip time between the client and the server, as measured by the observation point, calculated once per session. The value of this information element is the sum of all network delays observed for the sessions of this flow.<br>[2] [3] |
| collect connection delay network to-server sum | Field name: sumServerNwkTime<br>Server network delay is the round-trip time between the observation point and the server, calculated once per session. The value of this information element is the sum of all server network delays observed for the sessions of this flow.<br>[2] [3] |
| collect connection client counter packets retransmitted | Field name: retransClientPackets<br>Number of packets retransmitted by the client.<br>[2] [3] |
| collect connection server counter packets retransmitted | Field name: retransServerPackets<br>Number of packets retransmitted by the server.<br>[3] |

| Metric | Information |
|---|---|
| collect connection delay application sum | Field name: sumServerRespTime<br>The sum of all application delays observed for all responses of the flow.<br>[2] [3] |
| collect connection server counter responses | Field name: numRespsCountDelta<br>Total number of responses sent by the server.<br>[2] [3] |

**Notes**

[1] See Cisco IOS Flexible NetFlow Command Reference.

[2] See Cisco AVC Field Definition Guide.

[3] This metric can be used in Cisco Performance Monitor record types. It can be used with FNF only as part of the specially optimized Assurance-related records. Attempting to use this metric in a different FNF record type will cause the record to be rejected when attaching it to an interface.

# How to Configure Metrics for Assurance Monitoring

## Configuring Assurance Monitors Outside of DNA Center

In typical use, DNA Center configures the monitors without requiring additional user input, but it is possible to configure monitors for Assurance-related metrics manually.

Manual methods for monitoring Assurance-related metrics:

| Method | Applicable to... | See section... |
|---|---|---|
| ezPM profile | Platforms that support ezPM<br>Not wireless controllers | Configuring Assurance Monitors Using ezPM, on page 99 |
| Pre-defined FNF records for Assurance | Routers<br>Wireless controllers | Configuring Assurance Monitors Using Pre-defined FNF Records, on page 100 |

## Configuring Assurance Monitors Using ezPM

Applicable to: routers, not wireless controllers

The application-assurance ezPM profile makes use of the application performance monitoring (APM) FNF records designed for Assurance-related metrics. Configuring APM with ezPM greatly simplifies the configuration, as compared with working with the FNF records directly.

1. Configure the ezPM context.

   **performance monitor context** *context-name* **profile application-assurance**

   **traffic-monitor assurance-monitor ipv4**

**traffic-monitor assurance-monitor ipv6**

2. Attach the context to an interface. The following attaches the performance monitor to an interface, monitoring both input and output.

   **interface** *interface*

   **performance monitor context** *context-name*

### Result

This attaches monitors to the interface to collect Assurance-related metrics.

### Example

In the following example, a monitor called apm is attached to the Gigabit Ethernet 1 interface.

```
performance monitor context apm profile application-assurance
traffic-monitor assurance-monitor ipv4
traffic-monitor assurance-monitor ipv6

interface GigabitEthernet1
performance monitor context apm
```

# Configuring Assurance Monitors Using Pre-defined FNF Records

Applicable to: routers, wireless controllers

ezPM is the preferred method for configuring monitors for Assurance-related metrics, but it is also possible to use the FNF records pre-defined for these metrics. For platforms that do not support ezPM, this is the preferred method.

The FNF records designed for Assurance-related metrics are specially optimized for improved performance.

## How to configure on a routing platform

**Note** Does not apply to wireless platforms.

1. Define two flow monitors for assurance-related metrics, one for IPv4 and one for IPv6.

   **flow monitor** *monitor-name-for-ipv4*

   **cache entries 100000** {Optional. Recommended value depends on platform.}

   **record netflow ipv4 assurance**

   **flow monitor** *monitor-name-for-ipv6*

   **cache entries 100000** {Optional. Recommended value depends on platform.}

   **record netflow ipv6 assurance**

2. Attach the context to an interface. The following attaches the performance monitor to an interface, monitoring both input and output.

   **interface** *interface*

**ipv4 flow monitor** *monitor-name-for-ipv4* **input**

**ipv4 flow monitor** *monitor-name-for-ipv4* **output**

**ipv6 flow monitor** *monitor-name-for-ipv6* **input**

**ipv6 flow monitor** *monitor-name-for-ipv6* **output**

### Result

This attaches two IPv4 and two IPv6 monitors to the interface for collecting the metrics that are needed for Assurance.

### Example

This example defines monitors called assurance-ipv4 and assurance-ipv6, and attaches the monitors to the GigabitEthernet1 interface.

```
flow monitor assurance-ipv4
cache entries 100000
record netflow ipv4 assurance

flow monitor assurance-ipv6
cache entries 100000
record netflow ipv6 assurance

interface GigabitEthernet1
ipv4 flow monitor assurance-ipv4 input
ipv4 flow monitor assurance-ipv4 output
ipv6 flow monitor assurance-ipv6 input
ipv6 flow monitor assurance-ipv6 output
```

## How to configure on a wireless platform

**Note** Does not apply to routing platforms.

1. Enter the configuration mode for the relevant wireless profile.

   **interface** *policy-name*

2. Define two monitors for the wireless controller, one for IPv4 and one for IPv6.

   **flow monitor** *monitor-name-wlc-for-ipv4*

   **cache entries 100000** {Optional. Recommended value depends on platform.}

   **record wireless avc ipv4 assurance**

   **flow monitor monitor-name-wlc-ipv6**

   **cache entries 100000** {Optional. Recommended value depends on platform.}

   **record wireless avc ipv6 assurance**

3. Attach the two flow monitors to the wireless profile, including input and output traffic.

   **wireless profile policy** *policy-name*

**ipv4 flow monitor** *monitor-name-for-wireless-ipv4* **input**

**ipv4 flow monitor** *monitor-name-for-wireless-ipv4* **output**

**ipv6 flow monitor** monitor-name-for-wireless-ipv6 **input**

**ipv6 flow monitor** *monitor-name-for-wireless-ipv6* **output**

### Example

This example defines monitors called assurance-wlc-ipv4 and assurance-wlc-ipv6, and attaches the monitors to a wireless profile.

```
flow monitor assurance-wlc-ipv4
cache entries 100000
record wireless avc ipv4 assurance

flow monitor assurance-wlc-ipv6
cache entries 100000
record wireless avc ipv6 assurance

wireless profile policy AVC_POL
central association
central switching
ipv4 flow monitor assurance-wlc-ipv4 input
ipv4 flow monitor assurance-wlc-ipv4 output
ipv6 flow monitor assurance-wlc-ipv6 input
ipv6 flow monitor assurance-wlc-ipv6 output
no shutdown
```

# About Attaching the Assurance Monitors to Interfaces

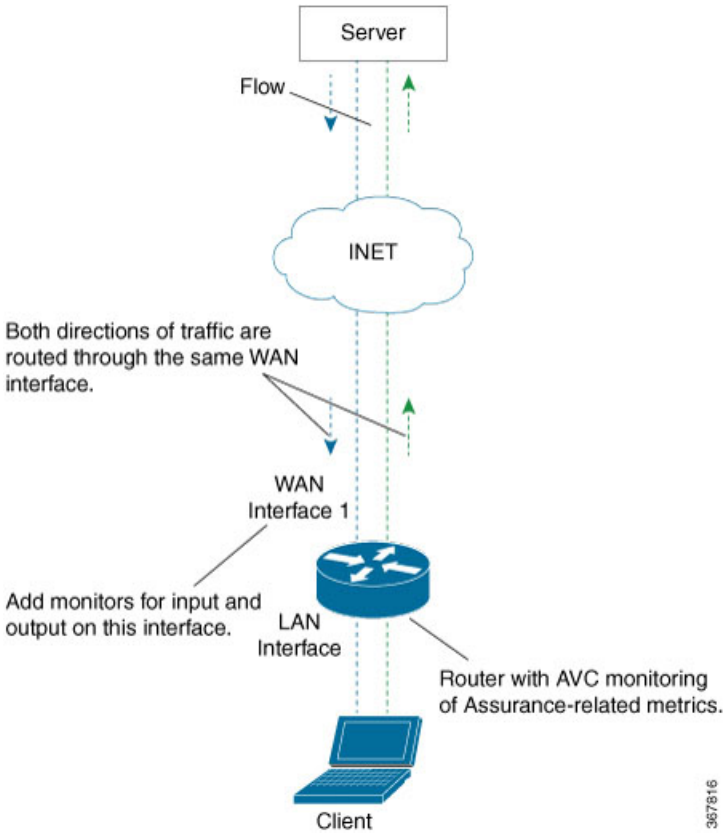
### Monitor a Flow on Only One Interface

Monitors for Assurance-related metrics should only see a single flow one time. In the typical symmetric routing scenario, they should monitor the flow on only one interface.

Do not attach monitors for Assurance-related metrics to two separate interfaces that handle both directions of the same flow. Doing so will cause incorrect traffic metrics to be reported. For example, if traffic enters a device on interface A and leaves on interface B, do not attach monitors for Assurance-related metrics to both interfaces A and B.

Typical symmetric routing, with monitors for input and output on the same interface:

*Figure 2: Symmetric Routing*

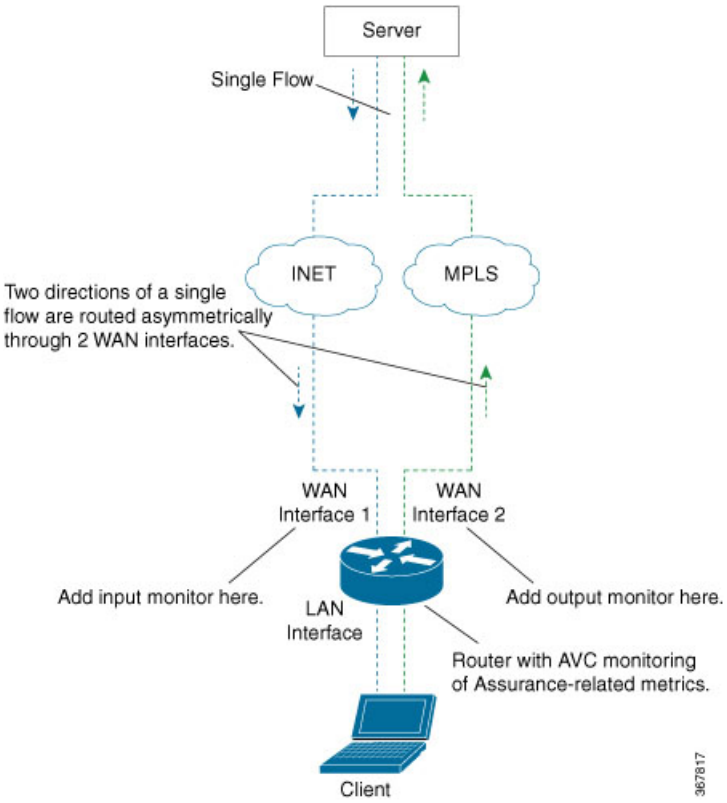



## Asymmetric Routing

In some cases, such as for asymmetric routing, it might be necessary to attach a monitor for input on one interface, and a monitor for output on another interface.

In some scenarios, a single flow may be routed asymmetrically, with upstream and downstream traffic for the flow occurring on two different interfaces. In this case, place monitors for input and output on two separate interfaces to monitor the complete flow.

*Figure 3: Asymmetric Routing*




# Viewing Details of Assurance Records and Contexts

## Overview

After you attach a context to an interface, two **show** commands can be used to display information about Assurance records or about contexts.

## Displaying Structure of the Assurance Record

The following command displays the structure of the pre-defined Assurance records (IPv4 and IPv6):

**show fnf record netflow** {**ipv4** | **ipv6**} **assurance**

## Displaying Configuration of a Context

The following command displays the full configuration of a specified context.

**show performance monitor context** *context-name* **configuration**

The following output shows the Assurance-related monitoring through an ezPM context called ApmContext, attached to a router interface.

```
Device#show performance monitor context ApmContext configuration
!==============================================================================
!                Equivalent Configuration of Context ApmContext               !
!==============================================================================
!Exporters
!==========
!
flow exporter ApmContext-1
description performance monitor context ApmContext exporter
destination 64.103.113.128 vrf FNF
source GigabitEthernet2/2/0
transport udp 2055
export-protocol ipfix
template data timeout 300
option interface-table timeout 300
option vrf-table timeout 300
option sampler-table timeout 300
option application-table timeout 300
option application-attributes timeout 300
!
!Access Lists
!=============
!Class-maps
!===========
!Samplers
!=========
!Records and Monitors
!=====================
!
flow record ApmContext-app_assurance_ipv4
description ezPM record
match ipv4 version
match ipv4 protocol
match application name
match connection client ipv4 address
match connection server ipv4 address
match connection server transport port
match flow observation point
collect routing vrf input
collect flow direction
collect timestamp absolute first
collect timestamp absolute last
collect connection initiator
collect connection new-connections
collect connection server counter responses
collect connection delay network to-server sum
collect connection client counter packets retransmitted
collect connection delay network client-to-server sum
collect connection delay application sum
collect connection server counter packets long
collect connection client counter packets long
collect connection server counter packets retransmitted
collect connection server counter bytes network long
collect connection client counter bytes network long
!
!
flow monitor ApmContext-app_assurance_ipv4
description ezPM monitor
exporter ApmContext-1
cache timeout active 60
cache entries 100000
record ApmContext-app_assurance_ipv4
!
!
```

```
flow record ApmContext-app_assurance_ipv6
description ezPM record
match ipv6 version
match ipv6 protocol
match application name
match connection client ipv6 address
match connection server transport port
match connection server ipv6 address
match flow observation point
collect routing vrf input
collect flow direction
collect timestamp absolute first
collect timestamp absolute last
collect connection initiator
collect connection new-connections
collect connection server counter responses
collect connection delay network to-server sum
collect connection client counter packets retransmitted
collect connection delay network client-to-server sum
collect connection delay application sum
collect connection server counter packets long
collect connection client counter packets long
collect connection server counter packets retransmitted
collect connection server counter bytes network long
collect connection client counter bytes network long
!
!
flow monitor ApmContext-app_assurance_ipv6
description ezPM monitor
exporter ApmContext-1
cache timeout active 60
cache entries 100000
record ApmContext-app_assurance_ipv6
!
!Interface Attachments
!=====================
interface TenGigabitEthernet2/0/0
ip flow monitor ApmContext-app_assurance_ipv4 input
ip flow monitor ApmContext-app_assurance_ipv4 output
ipv6 flow monitor ApmContext-app_assurance_ipv6 input
ipv6 flow monitor ApmContext-app_assurance_ipv6 output
```

# Notes and Limitations

## Assurance-related Metrics and Elephant Flows

In networking, especially long flows are termed, “elephant flows,” and can pose a challenge to networking resources.

In a case where a single high-burst flow consumes too many QFP resources, the monitor collecting Assurance metrics might stop collecting qualitative metrics for the flow, to preserve resources for other traffic. No other traffic is affected.

Quantitative metrics are collected fully:

- Flow packets start time
- Flow packet end time

- Packets
- Bytes

Qualitative metrics are not collected fully:

- Total network delay sum (in the TCP handshake)
- Network to-server delay sum (in the TCP handshake)
- Client packets retransmitted
- Server packets retransmitted
- Application delay sum
- Number of server application responses