# QoS: Policing and Shaping Configuration Guide, Cisco IOS Release 15M&T

**First Published:** January 22, 2013

**Last Modified:** January 22, 2013

**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 3

CHAPTER 4

CHAPTER 5

CHAPTER 16

CHAPTER 17

CHAPTER 18

CHAPTER 1

# Policing and Shaping Overview

Cisco IOS QoS offers two kinds of traffic regulation mechanisms--policing and shaping.

The rate-limiting features of committed access rate (CAR) and the Traffic Policing feature provide the functionality for policing traffic. The features of Generic Traffic Shaping (GTS), Class-Based Traffic Shaping, Distributed Traffic Shaping (DTS), and Frame Relay Traffic Shaping (FRTS) provide the functionality for shaping traffic.

You can deploy these features throughout your network to ensure that a packet, or data source, adheres to a stipulated contract and to determine the QoS to render the packet. Both policing and shaping mechanisms use the traffic descriptor for a packet--indicated by the classification of the packet--to ensure adherence and service.

Policers and shapers usually identify traffic descriptor violations in an identical manner. They usually differ, however, in the way they respond to violations, for example:

- A policer typically drops traffic. (For example, the CAR rate-limiting policer will either drop the packet or rewrite its IP precedence, resetting the type of service bits in the packet header.)
- A shaper typically delays excess traffic using a buffer, or queueing mechanism, to hold packets and shape the flow when the data rate of the source is higher than expected. (For example, GTS and Class-Based Shaping use a weighted fair queue to delay packets in order to shape the flow, and DTS and FRTS use either a priority queue, a custom queue, or a FIFO queue for the same, depending on how you configure it.)

Traffic shaping and policing can work in tandem. For example, a good traffic shaping scheme should make it easy for nodes inside the network to detect misbehaving flows. This activity is sometimes called policing the traffic of the flow.

This module gives a brief description of the Cisco IOS QoS traffic policing and shaping mechanisms. Because policing and shaping all use the token bucket mechanism, this module first explains how a token bucket works. This module includes the following sections:

# 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 at the end of this module.

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.

# What Is a Token Bucket

A token bucket is a formal definition of a rate of transfer. It has three components: a burst size, a mean rate, and a time interval (Tc). Although the mean rate is generally represented as bits per second, any two values may be derived from the third by the relation shown as follows:

```
mean rate = burst size / time interval
```

Here are some definitions of these terms:

- Mean rate--Also called the committed information rate (CIR), it specifies how much data can be sent or forwarded per unit time on average.
- Burst size--Also called the Committed Burst (Bc) size, it specifies in bits (or bytes) per burst, how much traffic can be sent within a given unit of time to not create scheduling concerns. (For a shaper, such as GTS, it specifies bits per burst; for a policer, such as CAR, it specifies bytes per burst, per second.)
- Time interval--Also called the measurement interval, it specifies the time quantum in seconds per burst.

By definition, over any integral multiple of the interval, the bit rate of the interface will not exceed the mean rate. The bit rate, however, may be arbitrarily fast within the interval.

A token bucket is used to manage a device that regulates the data in a flow. For example, the regulator might be a traffic policer, such as CAR, or a traffic shaper, such as FRTS or GTS. A token bucket itself has no discard or priority policy. Rather, a token bucket discards tokens and leaves to the flow the problem of managing its transmission queue if the flow overdrives the regulator. (Neither CAR nor FRTS and GTS implement either a true token bucket or true leaky bucket.)

In the token bucket metaphor, tokens are put into the bucket at a certain rate. The bucket itself has a specified capacity. If the bucket fills to capacity, newly arriving tokens are discarded. Each token is permission for the source to send a certain number of bits into the network. To send a packet, the regulator must remove from the bucket a number of tokens equal in representation to the packet size.

If not enough tokens are in the bucket to send a packet, the packet either waits until the bucket has enough tokens (in the case of GTS) or the packet is discarded or marked down (in the case of CAR). If the bucket is already full of tokens, incoming tokens overflow and are not available to future packets. Thus, at any time, the largest burst a source can send into the network is roughly proportional to the size of the bucket.

Note that the token bucket mechanism used for traffic shaping has both a token bucket and a data buffer, or queue; if it did not have a data buffer, it would be a policer. For traffic shaping, packets that arrive that cannot be sent immediately are delayed in the data buffer.

For traffic shaping, a token bucket permits burstiness but bounds it. It guarantees that the burstiness is bounded so that the flow will never send faster than the token bucket's capacity, divided by the time interval, plus the established rate at which tokens are placed in the token bucket. See the following formula:

```
(token bucket capacity in bits / time interval in seconds) + established rate in bps =
maximum flow speed in bps
```

This method of bounding burstiness also guarantees that the long-term transmission rate will not exceed the established rate at which tokens are placed in the bucket.

# Policing with CAR

Committed access rate (CAR) embodies a rate-limiting feature for policing traffic, in addition to its packet classification feature discussed in the "Classification Overview" module. The rate-limiting feature of CAR manages the access bandwidth policy for a network by ensuring that traffic falling within specified rate parameters is sent, while dropping packets that exceed the acceptable amount of traffic or sending them with a different priority. The exceed action for CAR is to drop or mark down packets.

The rate-limiting function of CAR does the following:

- Allows you to control the maximum rate of traffic sent or received on an interface.
- Gives you the ability to define Layer 3 aggregate or granular incoming or outgoing (ingress or egress) bandwidth rate limits and to specify traffic handling policies when the traffic either conforms to or exceeds the specified rate limits.

Aggregate bandwidth rate limits match all of the packets on an interface or subinterface. Granular bandwidth rate limits match a particular type of traffic based on precedence, MAC address, or other parameters.

CAR is often configured on interfaces at the edge of a network to limit traffic into or out of the network.

## How CAR Works

CAR examines traffic received on an interface or a subset of that traffic selected by access list criteria. It then compares the rate of the traffic to a configured token bucket and takes action based on the result. For example, CAR will drop the packet or rewrite the IP precedence by resetting the type of service (ToS) bits. You can configure CAR to send, drop, or set precedence.

CAR utilizes a token bucket measurement. Tokens are inserted into the bucket at the committed rate. The depth of the bucket is the burst size. Traffic arriving at the bucket when sufficient tokens are available is said to conform, and the corresponding number of tokens are removed from the bucket. If a sufficient number of tokens are not available, then the traffic is said to exceed.

### Matching Criteria

Traffic matching entails identification of traffic of interest for rate limiting, precedence setting, or both. Rate policies can be associated with one of the following qualities:

- Incoming interface
- All IP traffic
- IP precedence (defined by a rate-limit access list)

- MAC address (defined by a rate-limit access list)
- Multiprotocol Label Switching (MPLS) experimental (EXP) value (defined by a rate-limit access list)
- IP access list (standard and extended)

CAR provides configurable actions, such as send, drop, or set precedence when traffic conforms to or exceeds the rate limit.

**Note** Matching to IP access lists is more processor intensive than matching based on other criteria.

## Rate Limits

CAR propagates bursts. It does no smoothing or shaping of traffic, and therefore does no buffering and adds no delay. CAR is highly optimized to run on high-speed links--DS3, for example--in distributed mode on Versatile Interface Processors (VIPs) on the Cisco 7500 series.

CAR rate limits may be implemented either on input or output interfaces or subinterfaces including Frame Relay and ATM subinterfaces.

### What Rate Limits Define

Rate limits define which packets conform to or exceed the defined rate based on the following three parameters:

- Average rate. The average rate determines the long-term average transmission rate. Traffic that falls under this rate will always conform.
- Normal burst size. The normal burst size determines how large traffic bursts can be before some traffic exceeds the rate limit.
- Excess Burst size. The Excess Burst (Be) size determines how large traffic bursts can be before all traffic exceeds the rate limit. Traffic that falls between the normal burst size and the Excess Burst size exceeds the rate limit with a probability that increases as the burst size increases.

The maximum number of tokens that a bucket can contain is determined by the normal burst size configured for the token bucket.

When the CAR rate limit is applied to a packet, CAR removes from the bucket tokens that are equivalent in number to the byte size of the packet. If a packet arrives and the byte size of the packet is greater than the number of tokens available in the standard token bucket, extended burst capability is engaged if it is configured.

### Extended Burst Value

Extended burst is configured by setting the extended burst value greater than the normal burst value. Setting the extended burst value equal to the normal burst value excludes the extended burst capability. If extended burst is not configured, given the example scenario, the exceed action of CAR takes effect because a sufficient number of tokens are not available.

When extended burst is configured and this scenario occurs, the flow is allowed to borrow the needed tokens to allow the packet to be sent. This capability exists so as to avoid tail-drop behavior, and, instead, engage behavior like that of Random Early Detection (RED).

### How Extended Burst Capability Works

Here is how the extended burst capability works. If a packet arrives and needs to borrow *n* number of tokens because the token bucket contains fewer tokens than its packet size requires, then CAR compares the following two values:

- Extended burst parameter value.
- Compounded debt. Compounded debt is computed as the sum over all *ai*:
    - *a* indicates the actual debt value of the flow after packet *i* is sent. Actual debt is simply a count of how many tokens the flow has currently borrowed.
    - *i* indicates the *i*th packet that attempts to borrow tokens since the last time a packet was dropped.

If the compounded debt is greater than the extended burst value, the exceed action of CAR takes effect. After a packet is dropped, the compounded debt is effectively set to 0. CAR will compute a new compounded debt value equal to the actual debt for the next packet that needs to borrow tokens.

If the actual debt is greater than the extended limit, all packets will be dropped until the actual debt is reduced through accumulation of tokens in the token bucket.

Dropped packets do not count against any rate or burst limit. That is, when a packet is dropped, no tokens are removed from the token bucket.

**Note** Though it is true the entire compounded debt is forgiven when a packet is dropped, the actual debt is not forgiven, and the next packet to arrive to insufficient tokens is immediately assigned a new compounded debt value equal to the current actual debt. In this way, actual debt can continue to grow until it is so large that no compounding is needed to cause a packet to be dropped. In effect, at this time, the compounded debt is not really forgiven. This scenario would lead to excessive drops on streams that continually exceed normal burst. (See the example in the following section, "Actual and Compounded Debt Example, on page 5."

Testing of TCP traffic suggests that the chosen normal and extended burst values should be on the order of several seconds worth of traffic at the configured average rate. That is, if the average rate is 10 Mbps, then a normal burst size of 10 to 20 Mb and an Excess Burst size of 20 to 40 Mb would be appropriate.

### Recommended Burst Values

Cisco recommends the following values for the normal and extended burst parameters:

```
normal burst = configured rate * (1 byte)/(8 bits) * 1.5 seconds
extended burst = 2 * normal burst
```

With the listed choices for parameters, extensive test results have shown CAR to achieve the configured rate. If the burst values are too low, then the achieved rate is often much lower than the configured rate.

### Actual and Compounded Debt Example

This example shows how the compounded debt is forgiven, but the actual debt accumulates.

For this example, assume the following parameters:

- Token rate is 1 data unit per time unit

- Normal burst size is 2 data units
- Extended burst size is 4 data units
- 2 data units arrive per time unit

After 2 time units, the stream has used up its normal burst and must begin borrowing one data unit per time unit, beginning at time unit 3:

| Time | DU arrivals | Actual Debt | Compounded Debt |
|---|---|---|---|
| 1 | 2 | 0 | 0 |
| 2 | 2 | 0 | 0 |
| 3 | 2 | 1 | 1 |
| 4 | 2 | 2 | 3 |
| 5 | 2 | 3 (temporary) | 6 (temporary) |

At this time a packet is dropped because the new compounded debt (6) would exceed the extended burst limit (4). When the packet is dropped, the compounded debt effectively becomes 0, and the actual debt is 2. (The values 3 and 6 were only temporary and do not remain valid in the case where a packet is dropped.) The final values for time unit 5 follow. The stream begins borrowing again at time unit 6.

| Time | DU arrivals | Actual Debt | Compounded Debt |
|---|---|---|---|
| 5 | 2 | 2 | 0 |
| 6 | 2 | 3 | 3 |
| 7 | 2 | 4 (temporary) | 7 (temporary) |

At time unit 6, another packet is dropped and the debt values are adjusted accordingly.

| Time | DU arrivals | Actual Debt | Compounded Debt |
|---|---|---|---|
| 7 | 2 | 3 | 0 |

## Conform and Exceed Actions

CAR utilizes a token bucket, thus CAR can pass temporary bursts that exceed the rate limit as long as tokens are available.

Once a packet has been classified as conforming to or exceeding a particular rate limit, the router performs one of the following actions on the packet:

- Transmit--The packet is sent.
- Drop--The packet is discarded.
- Set precedence and transmit--The IP Precedence (ToS) bits in the packet header are rewritten. The packet is then sent. You can use this action to either color (set precedence) or recolor (modify existing packet precedence) the packet.
- Continue--The packet is evaluated using the next rate policy in a chain of rate limits. If there is not another rate policy, the packet is sent.
- Set precedence and continue--Set the IP Precedence bits to a specified value and then evaluate the next rate policy in the chain of rate limits.

For VIP-based platforms, two more actions are possible:

- Set QoS group and transmit--The packet is assigned to a QoS group and sent.

- Set QoS group and continue--The packet is assigned to a QoS group and then evaluated using the next rate policy. If there is not another rate policy, the packet is sent.

### Multiple Rate Policies

A single CAR rate policy includes information about the rate limit, conform actions, and exceed actions. Each interface can have multiple CAR rate policies corresponding to different types of traffic. For example, low priority traffic may be limited to a lower rate than high priority traffic. When there are multiple rate policies, the router examines each policy in the order entered until the packet matches. If no match is found, the default action is to send.

Rate policies can be independent: each rate policy deals with a different type of traffic. Alternatively, rate policies can be cascading: a packet may be compared to multiple different rate policies in succession.

Cascading of rate policies allows a series of rate limits to be applied to packets to specify more granular policies (for example, you could rate limit total traffic on an access link to a specified subrate bandwidth and then rate limit World Wide Web traffic on the same link to a given proportion of the subrate limit) or to match packets against an ordered sequence of policies until an applicable rate limit is encountered (for example, rate limiting several MAC addresses with different bandwidth allocations at an exchange point). You can configure up to a 100 rate policies on a subinterface.

## Restrictions of CAR and VIP-Distributed CAR

CAR and VIP-distributed CAR can only be used with IP traffic. Non-IP traffic is not rate limited.

CAR or VIP-distributed CAR can be configured on an interface or subinterface. However, CAR and VIP-distributed CAR are not supported on the following interfaces:

- Fast EtherChannel
- Tunnel
- PRI
- Any interface that does not support Cisco Express Forwarding (CEF)

CAR is only supported on ATM subinterfaces with the following encapsulations: aal5snap, aal5mux, and aal5nlpid.

**Note** CAR provides rate limiting and does not guarantee bandwidth. CAR should be used with other QoS features, such as distributed weighted fair queueing (DWFQ), if premium bandwidth assurances are required.

# Traffic Policing

Traffic policing allows you to control the maximum rate of traffic sent or received on an interface, and to partition a network into multiple priority levels or class of service (CoS).

The Traffic Policing feature manages the maximum rate of traffic through a token bucket algorithm. The token bucket algorithm can use the user-configured values to determine the maximum rate of traffic allowed on an interface at a given moment in time. The token bucket algorithm is affected by all traffic entering or leaving (depending on where the traffic policy with Traffic Policing configured) and is useful in managing network bandwidth in cases where several large packets are sent in the same traffic stream.

The token bucket algorithm provides users with three actions for each packet: a conform action, an exceed action, and an optional violate action. Traffic entering the interface with Traffic Policing configured is placed into one of these categories. Within these three categories, users can decide packet treatments. For instance, packets that conform can be configured to be transmitted, packets that exceed can be configured to be sent with a decreased priority, and packets that violate can be configured to be dropped.

Traffic Policing is often configured on interfaces at the edge of a network to limit the rate of traffic entering or leaving the network. In the most common Traffic Policing configurations, traffic that conforms is transmitted and traffic that exceeds is sent with a decreased priority or is dropped. Users can change these configuration options to suit their network needs.

The Traffic Policing feature supports the following MIBs:

- CISCO-CLASS-BASED-QOS-MIB
- CISCO-CLASS-BASED-QOS-CAPABILITY-MIB

This feature also supports RFC 2697, *A Single Rate Three Color Marker*.

For information on how to configure the Traffic Policing feature, see the "Configuring Traffic Policing" module.

# Benefits of Traffic Policing

### Bandwidth Management Through Rate Limiting

Traffic policing allows you to control the maximum rate of traffic sent or received on an interface. Traffic policing is often configured on interfaces at the edge of a network to limit traffic into or out of the network. Traffic that falls within the rate parameters is sent, whereas traffic that exceeds the parameters is dropped or sent with a different priority.

### Packet Marking Through IP Precedence, QoS Group, and DSCP Value Setting

Packet marking allows you to partition your network into multiple priority levels or classes of service (CoS), as follows:

- Use traffic policing to set the IP precedence or differentiated services code point (DSCP) values for packets entering the network. Networking devices within your network can then use the adjusted IP Precedence values to determine how the traffic should be treated. For example, the DWRED feature uses the IP Precedence values to determine the probability that a packet will be dropped.
- Use traffic policing to assign packets to a QoS group. The router uses the QoS group to determine how to prioritize packets.

# Restrictions for Traffic Policing

The following restrictions apply to the Traffic Policing feature:

- On a Cisco 7500 series router, traffic policing can monitor CEF switching paths only. In order to use the Traffic Policing feature, CEF must be configured on both the interface receiving the packet and the interface sending the packet.
- On a Cisco 7500 series router, traffic policing cannot be applied to packets that originated from or are destined to a router.
- Traffic policing can be configured on an interface or a subinterface.
- Traffic policing is not supported on the following interfaces:
    - Fast EtherChannel
    - Tunnel
    - PRI
    - Any interface on a Cisco 7500 series router that does not support CEF

## Prerequisites for Traffic Policing

On a Cisco 7500 series router, CEF must be configured on the interface before traffic policing can be used.

# Traffic Shaping to Regulate Packet Flow

Regulating the packet flow (that is, the flow of traffic) on the network is also known as traffic shaping. Traffic shaping allows you to control the speed of traffic leaving an interface. This way, you can match the flow of the traffic to the speed of the interface receiving the packet.

Cisco provides three mechanisms for regulating or shaping traffic: Class-Based Traffic Shaping, Generic Traffic Shaping (GTS), and Frame Relay Traffic Shaping (FRTS).

For more information about traffic shaping, see the "Regulating Packet Flow Using Traffic Shaping" module.

For information on configuring Frame Relay and FRTS, see the "Configuring Frame Relay" module and the "MQC-Based Frame Relay Traffic Shaping" module, respectively.

CHAPTER 2

# IPv6 QoS: MQC Traffic Policing

Configuration or command usage for policing are the same in IPv6 environments as for IPv4 environments.

# 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 at the end of this module.

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 IPv6 QoS: MQC Traffic Policing

## Implementation Strategy for QoS for IPv6

IPv6 packets are forwarded by paths that are different from those for IPv4. QoS features supported for IPv6 environments include packet classification, queueing, traffic shaping, weighted random early detection (WRED), class-based packet marking, and policing of IPv6 packets. These features are available at both the process switching and Cisco Express Forwarding switching paths of IPv6.

All of the QoS features available for IPv6 environments are managed from the modular QoS command-line interface (MQC). The MQC allows you to define traffic classes, create and configure traffic policies (policy maps), and then attach those traffic policies to interfaces.

To implement QoS in networks running IPv6, follow the same steps that you would follow to implement QoS in networks running only IPv4. At a very high level, the basic steps for implementing QoS are as follows:

- Know which applications in your network need QoS.
- Understand the characteristics of the applications so that you can make decisions about which QoS features would be appropriate.
- Know your network topology so that you know how link layer header sizes are affected by changes and forwarding.
- Create classes based on the criteria you establish for your network. In particular, if the same network is also carrying IPv4 traffic along with IPv6, decide if you want to treat both of them the same way or treat them separately and specify match criteria accordingly. If you want to treat them the same, use match statements such as **match precedence**, **match dscp**, **set precedence**, and **set dscp**. If you want to treat them separately, add match criteria such as **match protocol ip** and **match protocol ipv6** in a match-all class map.
- Create a policy to mark each class.
- Work from the edge toward the core in applying QoS features.
- Build the policy to treat the traffic.
- Apply the policy.

## Traffic Policing in IPv6 Environments

Congestion management for IPv6 is similar to IPv4, and the commands used to configure queueing and traffic shaping features for IPv6 environments are the same commands as those used for IPv4. Traffic shaping allows you to limit the packet dequeue rate by holding additional packets in the queues and forwarding them as specified by parameters configured for traffic shaping features. Traffic shaping uses flow-based queueing by default. CBWFQ can be used to classify and prioritize the packets. Class-based policer and generic traffic shaping (GTS) or Frame Relay traffic shaping (FRTS) can be used for conditioning and policing traffic.

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| IPv6 addressing and connectivity | *IPv6 Configuration Guide* |
| Cisco IOS commands | Cisco IOS Master Commands List, All Releases |
| IPv6 commands | Cisco IOS IPv6 Command Reference |
| Cisco IOS IPv6 features | Cisco_IOS_IPv6_Feature_Mapping |

| Related Topic | Document Title |
|---|---|
| Traffic Policing | "Traffic Policing" module in the *QoS: Policing and Shaping Configuration Guide*. |

### Standards and RFCs

| Standard/RFC | Title |
|---|---|
| RFCs for IPv6 | IPv6 RFCs |

### MIBs

| MIB | MIBs Link |
|---|---|
| | 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 |

### 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 IPv6 QoS: MQC Traffic Policing

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 . An account on Cisco.com is not required.

*Table 1: Feature Information for IPv6 QoS: MQC Traffic Policing*

| Feature Name | Releases | Feature Information |
|---|---|---|
| IPv6 QoS: MQC Traffic Policing | 12.0(28)S<br>12.2(33)SRA<br>12.2(18)SXE2<br>12.2(13)T<br>12.3<br>12.3(2)T<br>12.4<br>12.4(2)T | Configuration or command usage for policing are the same in IPv6 environments as for IPv4 environments. |

CHAPTER 3

# Configuring Traffic Policing

This feature module describes the Traffic Policing feature. Traffic policing allows you to control the maximum rate of traffic transmitted or received on an interface.

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

## 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 at the end of this module.

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.

## Feature Overview

The Traffic Policing feature performs the following functions:

- Limits the input or output transmission rate of a class of traffic based on user-defined criteria

- Marks packets by setting the ATM Cell Loss Priority (CLP) bit, Frame Relay Discard Eligibility (DE) bit, IP precedence value, IP differentiated services code point (DSCP) value, MPLS experimental value, and Quality of Service (QoS) group.

The Traffic Policing feature is applied when you attach a traffic policy contain the Traffic Policing configuration to an interface. A traffic policy is configured using the Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC).

The table below lists the feature history.

***Table 2: Feature History***

| Cisco IOS Release | Enhancement |
|---|---|
| 12.1(5)T | This command was introduced for Cisco IOS Release 12.1 T. A new Traffic Policing algorithm was introduced. The **violate-action** option became available. This feature became available on Cisco 2600, 3600, 4500, 7200, and 7500 series routers. |
| 12.2(2)T | The **set-clp-transmit** option for the *action* argument was added to the **police** command. The **set-frde-transmit** option for the *action* argument was added to the **police** command. However, the **set-frde-transmit** option is not supported for Any Transport over Multiprotocol Label Switching (MPLS) (AToM) traffic in this release. The **set-mpls-exp-transmit** option for the *action* argument was added to the **police** command. |
| Cisco IOS | For information about feature support in Cisco IOS software, use Cisco Feature Navigator. |

# Benefits

### Bandwidth Management Through Rate Limiting

Traffic policing allows you to control the maximum rate of traffic transmitted or received on an interface. Traffic policing is often configured on interfaces at the edge of a network to limit traffic into or out of the network. In most Traffic Policing configurations, traffic that falls within the rate parameters is transmitted, whereas traffic that exceeds the parameters is dropped or transmitted with a different priority.

### Packet Marking

Packet marking allows you to partition your network into multiple priority levels or classes of service (CoS). A packet is marked and these markings can be used to identify and classify traffic for downstream devices. In some cases, such as ATM Cell Loss Priority (CLP) marking or Frame Relay Discard Eligibility (DE) marking, the marking is used to classify traffic.

- Use traffic policing to set the IP precedence or DSCP values for packets entering the network. Networking devices within your network can then use the adjusted IP precedence values to determine how the traffic should be treated. For example, the Weighted Random Early Detection (WRED) feature uses the IP precedence values to determine the probability that a packet will be dropped.
- Use traffic policing to assign packets to a QoS group. The router uses the QoS group to determine how to prioritize packets within the router.

Traffic can be marked without using the Traffic Policing feature. If you want to mark traffic but do not want to use Traffic Policing, see the "Marking Network Traffic" module.

### Packet Prioritization for Frame Relay Frames

The Traffic Policing feature allows users to mark the Frame Relay DE bit of the Frame Relay frame. The Frame Relay DE bit is one bit and, therefore, can be set to either 0 or 1. In congested environments, frames with the DE bit set to 1 are discarded before frames with the DE bit set to 0.

### Packet Prioritization for ATM Cells

The Traffic Policing feature allows users to mark the ATM CLP bit in ATM cells. The ATM CLP bit is used to prioritize packets in ATM networks. The ATM CLP bit is one bit and, therefore, can be set to either 0 or 1. In congested environments, cells with the ATM CLP bit set to 1 are discarded before cells with the ATM CLP bit set to 0.

## Restrictions

- On a Cisco 7500 series router, traffic policing can monitor Cisco Express Forwarding (CEF) switching paths only. In order to use the Traffic Policing feature, Cisco Express Forwarding must be configured on both the interface receiving the packet and the interface sending the packet.
- On a Cisco 7500 series router, traffic policing cannot be applied to packets that originated from or are destined to a router.
- Traffic policing can be configured on an interface or a subinterface.
- Traffic policing is not supported on the following interfaces:
    - Fast EtherChannel
    - Tunnel

**Note** Traffic policing is supported on tunnels that are using the Cisco generic routing encapsulation (GRE) tunneling protocol.

- 
    - PRI
    - Any interface on a Cisco 7500 series router that does not support Cisco Express Forwarding

# Supported Platforms

- Cisco 2500 series

**Note** Cisco IOS Release 12.2(2)T or later does not run on Cisco 2500 series routers.

- Cisco 2600 series
- Cisco 3640 routers
- Cisco 4500 series
- Cisco 7000 series with RSP7000
- Cisco 7100 series
- Cisco 7200 series
- Cisco 7500 series

**Note** To use the **set-clp-transmit** action available with this feature, the Enhanced ATM Port Adapter (PA-A3) is required. Therefore, the **set-clp-transmit**action is not supported on any platform that does not support the PA-A3 adapter (such as the Cisco 2600 series router, the Cisco 3640 router, and the 4500 series router). For more information, see the documentation for your specific router.

# Supported Standards MIBs and RFCs

### Standards

No new or modified standards are supported by this feature.

### MIBs

*Class-Based Quality of Service MIB*

- CISCO-CLASS-BASED-QOS-MIB
- CISCO-CLASS-BASED-QOS-CAPABILITY-MIB

For descriptions of supported MIBs and how to use MIBs, see the Cisco MIB web site on CCO at http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.

### RFCs

- RFC 2697, *A Single Rate Three Color Marker*

# Prerequisites

On a Cisco 7500 series router, Cisco Express Forwarding (CEF) must be configured on the interface before traffic policing can be used.

# Configuration Tasks

## Configuring Traffic Policing

| Command | Purpose |
|---|---|
| Router(config-pmap-c)# **police** *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action* **violate-action** *action* | Specifies a maximum bandwidth usage by a traffic class.<br>**Note** The Traffic Policing feature works with a token bucket mechanism. There are currently two types of token bucket algorithms: a single token bucket algorithm and a two token bucket algorithm. A single token bucket system is used when the **violate-action** option is not specified, and a two token bucket system is used when the **violate-action** option is specified. |

## Troubleshooting Tips

- Check the interface type. Verify that your interface is not mentioned in the nonsupported interface description in the Restrictions, on page 17 section of this module.
- For input traffic policing on a Cisco 7500 series router, verify that CEF is configured on the interface where traffic policing is configured.
- For output traffic policing on a Cisco 7500 series router, ensure that the incoming traffic is CEF-switched. Traffic policing cannot be used on the switching path unless CEF switching is enabled.

# Monitoring and Maintaining Traffic Policing

| Command | Purpose |
|---|---|
| Router# **show policy-map** | Displays all configured policy maps. |
| Router# **show policy-map** *policy-map-name* | Displays the user-specified policy map. |

| Command | Purpose |
|---|---|
| Router# **show policy-map interface** | Displays statistics and configurations of all input and output policies that are attached to an interface. |

# Configuration Examples

## Example Configuring a Service Policy that Includes Traffic Policing

The following configuration shows how to define a traffic class (with the **class-map**command) and associate that traffic class with a traffic policy (with the **policy-map** command). Traffic policing is applied in the traffic policy. The **service-policy** command is then used to attach the traffic policy to the interface.

In this example, traffic policing is configured with the average rate at 8000 bits per second, the normal burst size at 1500 bytes, and the excess burst size at 4000 bytes. Packets coming into Fast Ethernet interface 0/0 are evaluated by the token bucket algorithm to analyze whether packets conform, exceed, or violate the parameters. Packets that conform are transmitted, packets that exceed are assigned a QoS group value of 4 and are transmitted, packets that violate are dropped.

```
Router(config)# class-map acgroup2
Router(config-cmap)# match access-group 2
Router(config-cmap)# exit
Router(config)# policy-map police
Router(config-pmap)# class acgroup2
Router(config-pmap-c)# police 8000 1500 4000 conform-action transmit exceed-action
set-qos-transmit 4 violate-action drop
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface fastethernet 0/0
Router(config-if)# service-policy input police
```

CHAPTER 4

# IPv6 QoS: MQC Traffic Shaping

Traffic shaping allows you to limit the packet dequeue rate by holding additional packets in the queues and forwarding them as specified by parameters configured for traffic shaping features

## 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 at the end of this module.

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 IPv6 QoS: MQC Traffic Shaping

### Implementation Strategy for QoS for IPv6

IPv6 packets are forwarded by paths that are different from those for IPv4. QoS features supported for IPv6 environments include packet classification, queueing, traffic shaping, weighted random early detection (WRED), class-based packet marking, and policing of IPv6 packets. These features are available at both the process switching and Cisco Express Forwarding switching paths of IPv6.

All of the QoS features available for IPv6 environments are managed from the modular QoS command-line interface (MQC). The MQC allows you to define traffic classes, create and configure traffic policies (policy maps), and then attach those traffic policies to interfaces.

To implement QoS in networks running IPv6, follow the same steps that you would follow to implement QoS in networks running only IPv4. At a very high level, the basic steps for implementing QoS are as follows:

- Know which applications in your network need QoS.
- Understand the characteristics of the applications so that you can make decisions about which QoS features would be appropriate.
- Know your network topology so that you know how link layer header sizes are affected by changes and forwarding.
- Create classes based on the criteria you establish for your network. In particular, if the same network is also carrying IPv4 traffic along with IPv6, decide if you want to treat both of them the same way or treat them separately and specify match criteria accordingly. If you want to treat them the same, use match statements such as **match precedence**, **match dscp**, **set precedence**, and **set dscp**. If you want to treat them separately, add match criteria such as **match protocol ip** and **match protocol ipv6** in a match-all class map.
- Create a policy to mark each class.
- Work from the edge toward the core in applying QoS features.
- Build the policy to treat the traffic.
- Apply the policy.

## Traffic Policing in IPv6 Environments

Congestion management for IPv6 is similar to IPv4, and the commands used to configure queueing and traffic shaping features for IPv6 environments are the same commands as those used for IPv4. Traffic shaping allows you to limit the packet dequeue rate by holding additional packets in the queues and forwarding them as specified by parameters configured for traffic shaping features. Traffic shaping uses flow-based queueing by default. CBWFQ can be used to classify and prioritize the packets. Class-based policer and generic traffic shaping (GTS) or Frame Relay traffic shaping (FRTS) can be used for conditioning and policing traffic.

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| IPv6 addressing and connectivity | *IPv6 Configuration Guide* |
| Cisco IOS commands | Cisco IOS Master Commands List, All Releases |
| IPv6 commands | Cisco IOS IPv6 Command Reference |
| Cisco IOS IPv6 features | Cisco_IOS_IPv6_Feature_Mapping |

### Standards and RFCs

| Standard/RFC | Title |
|---|---|
| RFCs for IPv6 | IPv6 RFCs |

### MIBs

| MIB | MIBs Link |
|---|---|
| | 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 |

### 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 IPv6 QoS: MQC Traffic Shaping

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 . An account on Cisco.com is not required.

*Table 3: Feature Information for IPv6 QoS: MQC Traffic Shaping*

| Feature Name | Releases | Feature Information |
|---|---|---|
| IPv6 QoS: MQC Traffic Shaping | 12.2(13)T<br>12.3<br>12.2(50)SG<br>3.2.0SG<br>15.0(2)SG<br>12.2(33)SRA<br>12.2(18)SXE | Traffic shaping allows you to limit the packet dequeue rate by holding additional packets in the queues and forwarding them as specified by parameters configured for traffic shaping features. |

CHAPTER 5

# Two-Rate Policer

This document describes the Two-Rate Policer feature and how to configure it. Two-Rate Policer allows you to manage traffic rates through an interface; it is especially helpful in managing network bandwidth where large packets are in the same traffic stream.

## 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 at the end of this module.

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

## Prerequisites for Two-Rate Policer

### Supported Platforms

- Cisco 2600 series
- Cisco 3620

- Cisco 3640
- Cisco 7100 series
- Cisco 7200 series
- Cisco 7500 series (VIP-based platform only)

**Note** The **set-clp-transmit** action available with Two-Rate Policer, the Enhanced ATM Port Adapter (PA-A3) is required. The **set-clp-transmit** action is not supported on any platform that does not support the PA-A3 adapter (such as the Cisco 2600 series router, the Cisco 3620 router, and the 3640 router). For more information, see the documentation for your specific router.

- On a Cisco 7500 series router, Cisco Express Forwarding or Distributed Cisco Express Forwarding must be configured on the interface before you can use the Two-Rate Policer.
- A traffic class and a service policy must be created, and the service policy must be attached to a specified interface. These tasks are performed using the Modular quality of service (QoS) Command-Line Interface (CLI) (MQC). For information on the MQC, see the "Applying QoS Features Using the MQC" module.

# Restrictions for Two-Rate Policer

The following restrictions apply to the Two-Rate Policer feature:

- On a Cisco 7500 series router, traffic policing can monitor Cisco Express Forwarding or Distributed Cisco Express Forwarding switching paths only. Cisco Express Forwarding or Distributed Cisco Express Forwarding must be configured on both the interface receiving the packet and the interface sending the packet.
- On a Cisco 7500 series router, traffic policing cannot be applied to packets that originated from or are destined to a router.
- Two-rate policing can be configured on an interface, a subinterface, a Frame Relay data-link connection identifier (DLCI), and an ATM permanent virtual circuit (PVC).
- Two-rate policing is not supported on the following interfaces:
    - Fast EtherChannel
    - PRI
    - Any interface on a Cisco 7500 series router that does not support Cisco Express Forwarding or Distributed Cisco Express Forwarding

# Information About Two-Rate Policer

Networks police traffic by limiting the input or output transmission rate of a class of traffic based on user-defined criteria. Policing traffic allows you to control the maximum rate of traffic sent or received on an interface and to partition a network into multiple priority levels or class of service (CoS).

The Two-Rate Policer performs the following functions:

- Limits the input or output transmission rate of a class of traffic based on user-defined criteria.
- Marks packets by setting the IP precedence value, IP differentiated services code point (DSCP) value, Multiprotocol Label Switching (MPLS) experimental value, Quality of Service (QoS) group, ATM Cell Loss Priority (CLP) bit, and the Frame Relay Discard Eligibility (DE) bit.

With the Two-Rate Policer, you can enforce traffic policing according to two separate rates--committed information rate (CIR) and peak information rate (PIR). You can specify the use of these two rates, along with their corresponding values, by using two keywords, **cir** and **pir**, of the **police** command.

The Two-Rate Policer manages the maximum rate of traffic through a token bucket algorithm. The token bucket algorithm can use the user-configured values to determine the maximum rate of traffic allowed on an interface at a given moment in time. The token bucket algorithm is affected by all traffic entering or leaving the interface (depending on the location of the interface on which the Two-Rate Policer is configured) and is useful in managing network bandwidth in cases where several large packets are sent in the same traffic stream.

The token bucket algorithm provides users with three actions for each packet: a conform action, an exceed action, and an optional violate action. Traffic coming into the interface with the Two-Rate Policer configured is assigned one of these categories. Within these three categories, users can decide packet treatments. For instance, packets that conform can be configured to be sent, packets that exceed can be configured to be sent with a decreased priority, and packets that violate can be configured to be dropped.

The Two-Rate Policer is often configured on interfaces at the edge of a network to limit the rate of traffic entering or leaving the network. In the most common configurations, traffic that conforms is sent and traffic that exceeds is sent with a decreased priority or is dropped. Users can change these configuration options to suit their network needs.

**Note** Two-Rate Policer enables you to use Differentiated Services (DiffServ) Assured Forwarding (AF) Per-Hop Behavior (PHB) traffic conditioning. For more information about DiffServ, see the "Implementing DiffServ for End-to-End Quality of Service Overview" module.

**Note** Starting with Cisco IOS Release 12.1(5)T, you can police traffic by using the Traffic Policing feature (sometimes referred to as the single-rate policer). The Two-Rate Policer (available with Cisco IOS Release 12.2(4)T) is in addition to the Traffic Policing feature, and it provides additional functionality. For more information about the Traffic Policing feature, see the "Traffic Policing" module.

# Benefits

### Bandwidth Management Through Rate Limiting

Two-Rate Policer provides improved bandwidth management through rate limiting. Before this feature was available, you could police traffic with the single-rate Traffic Policing feature. The Traffic Policing feature provided a certain amount of bandwidth management by allowing you to set the peak burst size (be). The Two-Rate Policer supports a higher level of bandwidth management and supports a sustained excess rate. With the Two-Rate Policer, you can enforce traffic policing according to two separate rates--CIR and PIR--specified in bits per second (bps).

### Packet Marking Through IP Precedence, DSCP Value, MPLS Experimental Value, and the QoS Group Setting

In addition to rate-limiting, the Two-Rate Policer allows you to independently mark the packet according to whether the packet conforms, exceeds, or violates a specified rate. Packet marking also allows you to partition your network into multiple priority levels or CoSs.

- Use the Two-Rate Policer to set the IP precedence value, the IP DSCP value, or the MPLS experimental value for packets that enter the network. Then networking devices within your network can use this setting to determine how the traffic should be treated. For example, the Weighted Random Early Detection (WRED) feature uses the IP precedence value to determine the probability that a packet will be dropped.
- Use the Two-Rate Policer to assign packets to a QoS group. The router uses the QoS group to determine how to prioritize packets within the router.

If you want to mark traffic but do not want to use the Two-Rate Policer, see the "Marking Network Traffic" module.

### Packet Marking for Frame Relay Frames

The Two-Rate Policer allows users to mark the Frame Relay DE bit of the Frame Relay frame. The Frame Relay DE bit is one bit and, therefore, can be set to either 0 or 1. In congested environments, frames that have the DE bit set to 1 are discarded before frames that have the DE bit set to 0.

### Packet Marking for ATM Cells

The Two-Rate Policer allows users to mark the ATM CLP bit in ATM cells. The ATM CLP bit is used to prioritize packets in ATM networks. The ATM CLP bit is one bit and, therefore, can be set to either 0 or 1. In congested environments, cells that have the ATM CLP bit set to 1 are discarded before cells that have the ATM CLP bit set to 0.

# How to Use the Two-Rate Policer

## Configuring the Two-Rate Policer

| Command | Purpose |
|---|---|
| `Router(config-pmap-c)#`<br>**police cir** *cir* [**bc** *conform-burst*] **pir** *pir* [**be** *peak-burst* | Specifies that both the CIR and the PIR are to be used for two-rate traffic policing. The **bc** and **be** keywords and their associated arguments (*conform-burst* and *peak-burst*, respectively) are optional.<br>Specifies the action taken on a packet when you enable an optional **action** argument.<br>**Note** The Two-Rate Policer works by using a token bucket mechanism. There are currently two types of token bucket algorithms: a single token bucket algorithm (available through the Traffic Policing feature) and a two token bucket algorithm (available through the Two-Rate Policer). |

## Verifying the Two-Rate Policer Configuration

| Command | Purpose |
|---|---|
| `Router#`<br>**show policy-map interface** | Displays statistics and configurations of all input and output policies attached to an interface. |

## Troubleshooting Tips

- Check the interface type. Verify that your interface is not listed as a nonsupported interface in the Restrictions for Two-Rate Policer, on page 26 section of this module.
- For input traffic policing on a Cisco 7500 series router, verify that Cisco Express Forwarding or Distributed Cisco Express Forwarding is configured on the interface on which traffic policing is configured.
- For output traffic policing on a Cisco 7500 series router, ensure that the incoming traffic is Cisco Express Forwarding-switched or Distributed Cisco Express Forwarding-switched. Traffic policing cannot be used on the switching path unless Cisco Express Forwarding or Distributed Cisco Express Forwarding switching is enabled.

## Monitoring and Maintaining the Two-Rate Policer

| Command | Purpose |
|---|---|
| `Router#`<br>**show policy-map** | Displays all configured policy maps. |
| `Router#` **show policy-map** *policy-map-name* | Displays the user-specified policy map. |
| `Router#`<br>**show policy-map interface** | Displays statistics and configurations of all input and output policies that are attached to an interface. |

# Configuration Examples

## Example Limiting the Traffic Using a Policer Class

In this example, the Two-Rate Policer is configured on a class to limit traffic to an average committed rate of 500 kbps and a peak rate of 1 Mbps:

```
Router(config)# class-map
 police
Router(config-cmap)# match
 access-group 10
1
Router(config-cmap)# policy-map
 policy1
Router(config-pmap)# class
 police
Router(config-pmap-c)# police
 cir 500000 bc 10000 pir 1000000 be 10000 conform-action transmit exceed-action
set-prec-transmit 2 violate-action drop
Router(config)# interface
 serial3/0
Router(config-if)# service-policy
 output policy1
Router(config-if)# end
Router# show
 policy-map policy1
 Policy Map policy1
  Class police
   police cir 500000 conform-burst 10000 pir 1000000 peak-burst 10000 conform-action transmit
 exceed-action set-prec-transmit 2 violate-action drop
```

Traffic marked as conforming to the average committed rate (500 kbps) will be sent as is. Traffic marked as exceeding 500 kbps, but not exceeding 1 Mbps, will be marked with IP Precedence 2 and then sent. All traffic exceeding 1 Mbps will be dropped. The burst parameters are set to 10,000 bytes.

```
Router# show
 policy-map interface serial3/0
 Serial3/0
  Service-policy output: policy1
   Class-map: police (match all)
    148803 packets, 36605538 bytes
    30 second offered rate 1249000 bps, drop rate 249000 bps
    Match: access-group 101
    police:
     cir 500000 bps, conform-burst 10000, pir 1000000, peak-burst 100000
     conformed 59538 packets, 14646348 bytes; action: transmit
     exceeded 59538 packets, 14646348 bytes; action: set-prec-transmit 2
     violated 29731 packets, 7313826 bytes; action: drop
     conformed 499000 bps, exceed 500000 bps violate 249000 bps
   Class-map: class-default (match-any)
    19 packets, 1990 bytes
    30 seconds offered rate 0 bps, drop rate 0 bps
    Match: any
```

# Additional References

The following sections provide references related to the Two-Rate Policer feature.

### Related Documents

| Related Topic | Document Title |
|---|---|
| MQC | - "Applying QoS Features Using the MQC" module |
| QoS features such as class-based weighted fair queueing (CBWFQ), traffic marking, and traffic policing | - "Configuring Weighted Fair Queueing" module<br>- "Marking Network Traffic" module<br>- "Traffic Policing" module |

### Standards

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

### MIBs

| MIB | MIBs Link |
|---|---|
| - CISCO-CLASS-BASED-QOS-MIB<br>- CISCO-CLASS-BASED-QOS-CAPABILITY-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 2698 | *A Two Rate Three Color Marker* |

### 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/cisco/web/support/index.html |

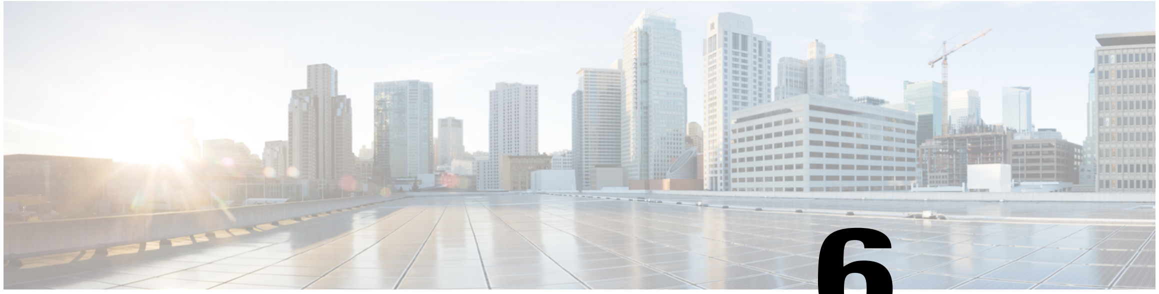
# Feature Information for Two-Rate Policer

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 . An account on Cisco.com is not required.

*Table 4: Feature Information for Two-Rate Policer*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Two-Rate Policer | 12.2(4)T<br>12.2(4)T3<br>12.0(26)S<br>12.2(28)SB<br>12.2(33)SRA<br>12.2(33)SXH<br>Cisco IOS XE Release 2.1<br>Cisco IOS XE 3.1.0 SG | This feature was introduced.<br>Support for the Cisco 7500 series routers was added.<br>This feature was integrated into Cisco IOS Release 12.0(26)S for the Cisco 7200 and 7500 series routers.<br>This feature was integrated into Cisco IOS Release 12.2(28)SB.<br>This feature was integrated into Cisco IOS Release 12.2(33)SRA.<br>This feature was integrated into Cisco IOS Release 12.2(33)SXH.<br>This feature was implemented on Cisco ASR 1000 Series Routers.<br>This feature was integrated into Cisco IOS XE 3.1.0 SG. |



CHAPTER 6

# Policer Enhancement - Multiple Actions

**Feature History**

| Release | Modification |
|---|---|
| 12.2(8)T | This feature was introduced. |

This document describes the Policer Enhancement -- Multiple Actions feature in Cisco IOS Release 12.2(8)T. It includes the following sections:

# 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 at the end of this module.

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.

# Feature Overview

This feature further extends the functionality of the Cisco IOS Traffic Policing feature (a single-rate policer) and the Two-Rate Policer feature. The Traffic Policing and Two-Rate Policer features are traffic policing mechanisms that allow you to control the maximum rate of traffic sent or received on an interface. Both of these traffic policing mechanisms mark packets as either conforming to, exceeding, or violating a specified rate. After a packet is marked, you can specify an action to be taken on the packet based on that marking.

With both the Traffic Policing feature and the Two-Rate Policer feature, you can specify only one conform action, one exceed action, and one violate action. Now with the new Policer Enhancement -- Multiple Actions feature, you can specify multiple conform, exceed, and violate actions for the marked packets.

## Benefits

Before this feature, you could specify only *one* marking action for a packet, in addition to transmitting the packet. This feature provides enhanced flexibility by allowing you to specify *multiple* marking actions for a packet, as required. For example, if you know the packet will be transmitted through both a TCP/IP and a Frame Relay environment, you can change the DSCP value of the exceeding or violating packet, and also set the Frame Relay Discard Eligibility (DE) bit from 0 to 1 to indicate lower priority.

## Restrictions

- On a Cisco 7500 series router, traffic policing can monitor Cisco Express Forwarding (CEF) or distributed CEF (dCEF) switching paths only. To use the Two-Rate Policer, CEF or dCEF must be configured on both the interface receiving the packet and the interface sending the packet.
- On a Cisco 7500 series router, traffic policing cannot be applied to packets that originated from or are destined to a router.
- Multiple policer actions can be configured on an interface, a subinterface, a Frame Relay data-link connection identifier (DLCI), and an ATM permanent virtual circuit (PVC) only.
- When using this feature, you can specify a maximum of four actions at one time.
- Multiple policer actions are not supported on the following interfaces:
    - Fast EtherChannel
    - PRI
    - Any interface on a Cisco 7500 series router that does not support CEF or dCEF

## Related Features and Technologies

- Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC)
- Class-Based Weighted Fair Queueing (CBWFQ)
- Class-Based Packet Marking

- Traffic Policing
- Two-Rate Policing

## Related Documents

- "Applying QoS Features Using the MQC" module
- "Configuring Weighted Fair Queueing" module
- Marking Network Traffic" module
- "Policing and Shaping Overview" module
- "Traffic Policing" module
- "Two-Rate Policer" module
- Cisco IOS Quality of Service Solutions Command Reference.
- RFC 2697, *A Single Rate Three Color Marker*
- RFC 2698, *A Two Rate Three Color Marker*

# Supported Platforms

- Cisco 1700 series
- Cisco 2600 series
- Cisco 3620
- Cisco 3640
- Cisco 3660
- Cisco 7100 series
- Cisco 7200 series
- Cisco 7500 series (VIP-based platform only)
- Cisco MC3810

**Note** To use the *set-clp-transmit*action available with this feature, the Enhanced ATM Port Adapter (PA-A3) is required. Therefore, the *set-clp-transmit* action is not supported on any platform that does not support the PA-A3 adapter (such as the Cisco 2600 series router and the Cisco 3640 router). For more information, refer to the documentation for your specific router.

# Supported Standards MIBs and RFCs

### Standards

No new or modified standards are supported by this feature.

### MIBs

- CISCO-CLASS-BASED-QOS-MIB
- CISCO-CLASS-BASED-QOS-CAPABILITY-MIB

To obtain lists of supported MIBs by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB website on Cisco.com at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

### RFCs

- RFC 2697, *A Single Rate Three Color Marker*
- RFC 2698, *A Two Rate Three Color Marker*

# Prerequisites

- On a Cisco 7500 series router, CEF or dCEF must be configured on the interface before you can use the Policer Enhancement -- Multiple Actions feature.
- To configure the Policer Enhancement -- Multiple Actions feature, a traffic class and a service policy must be created, and the service policy must be attached to a specified interface.

# Configuration Tasks

## Configuring Multiple Policer Actions

### SUMMARY STEPS

1. Router(config)# **policy-map** *policy-map-name*
2. Router(config-pmap)# **class** *class-default*
3. Router(config-pmap-c)# **police** {**cir** *cir*}[**bc** *conform-burst*]{**pir** *pir*} [**be** *peak-burst*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]]

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Router(config)# **policy-map** *policy-map-name* | Creates a policy map. Enters policy-map configuration mode. |
| **Step 2** | Router(config-pmap)# **class** *class-default* | Specifies the default traffic class for a service policy. Enters policy-map class configuration mode. |
| **Step 3** | Router(config-pmap-c)# **police** {**cir** *cir*}[**bc** *conform-burst*]{**pir** *pir*} [**be** *peak-burst*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]] | Configures traffic policing and specifies multiple actions applied to packets marked as conforming to, exceeding, or violating a specific rate. Use one line per action that you want to specify. Enters policy-map class police configuration mode. |

# Verifying the Multiple Policer Actions Configuration

| Command | Purpose |
|---|---|
| `Router#`<br>**show policy-map interface** | Displays statistics and configurations of all input and output policies attached to an interface. |

# Troubleshooting Tips

- Check the interface type. Verify that your interface is not listed as a nonsupported interface in the "Restrictions" section of this document.
- For input traffic policing on a Cisco 7500 series router, verify that CEF or dCEF is configured on the interface on which traffic policing is configured.
- For output traffic policing on a Cisco 7500 series router, ensure that the incoming traffic is CEF-switched or dCEF-switched. Traffic policing cannot be used on the switching path unless CEF or dCEF switching is enabled.

# Monitoring and Maintaining the Multiple Policer Actions

| Command | Purpose |
|---|---|
| `Router#` **show policy-map** | Displays all configured policy maps. |
| `Router#` **show policy-map** *policy-map-name* | Displays the user-specified policy map. |
| `Router#` **show policy-map interface** | Displays statistics and configurations of all input and output policies that are attached to an interface. |

# Configuration Examples

## Example Multiple Actions in a Two-Rate Policer

In the following example, a policy map called police is configured to use a two-rate policer to police traffic leaving an interface. Two rates, a committed information rate (CIR) of 1 Mbps and a peak information rate (PIR) of 2 Mbps, have been specified.

```
Router(config)# policy-map police
Router(config-pmap)# class class-default
Router(config-pmap-c)# police cir 1000000 pir 2000000

Router(config-pmap-c-police)# conform-action transmit
Router(config-pmap-c-police)# exceed-action set-prec-transmit 4
Router(config-pmap-c-police)# exceed-action set-frde
Router(config-pmap-c-police)# violate-action set-prec-transmit 2
Router(config-pmap-c-police)# violate-action set-frde-transmit

Router(config-pmap-c-police)# end
```

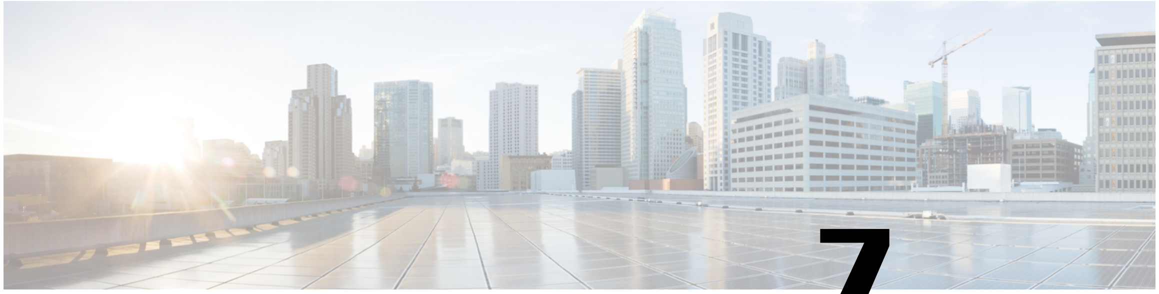
The following actions will be performed on packets associated with the policy map called police:

- All packets marked as conforming to these rates (that is, packets conforming to the CIR) will be transmitted unaltered.
- All packets marked as exceeding these rates (that is, packets exceeding the CIR but not exceeding the PIR) will be assigned an IP Precedence level of 4, the DE bit will be set to 1, and then transmitted.
- All packets marked as violating the rate (that is, exceeding the PIR) will be assigned an IP Precedence level of 2, the DE bit will be set to 1, and then transmitted.

## Example Verifying the Multiple Policer Actions

The following sample output of the **show policy-map**command displays the configuration for a service policy called police. In this service policy, multiple actions for packets marked as exceeding the specified CIR rate have been configured. For those packets, the IP Precedence level is set to 4, the DE bit is set to 1, and the packet is transmitted. Multiple actions for packets marked as violating the specified PIR rate have also been configured. For those packets, the IP Precedence level is set to 2, the DE bit is set to 1, and the packet is transmitted.

```
Router# show policy-map police
  Policy Map police
    Class class-default
     police cir 1000000 bc 31250 pir 2000000 be 31250
       conform-action transmit
       exceed-action set-prec-transmit 4
       exceed-action set-frde-transmit
       violate-action set-prec-transmit 2
       violate-action set-frde-transmit
```



CHAPTER 7

# Percentage-Based Policing and Shaping

### Feature History

| Release | Modification |
|---|---|
| 12.2(13)T | This feature was introduced. |
| Supported Platforms | |
| For platforms supported in Cisco IOS Release 12.2(13)T, consult Cisco Feature Navigator. | |

This document describes the Percentage-Based Policing and Shaping feature in Cisco IOS Release 12.2(13)T. It includes the following sections:

## 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 at the end of this module.

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.

# Feature Overview

Cisco IOS quality of service (QoS) offers two kinds of traffic regulation mechanisms--traffic policing and traffic shaping. A traffic policer typically drops traffic that violates a specific rate. A traffic shaper typically delays excess traffic using a buffer to hold packets and shapes the flow when the data rate to a queue is higher than expected.

Traffic shaping and traffic policing can work in tandem and can be configured in a class map. Class maps organize data packets into specific categories ("classes") that can, in turn, receive a user-defined QoS treatment when used in policy maps (sometimes referred to as "service policies").

Before this feature, traffic policing and traffic shaping were configured on the basis of a user-specified amount of bandwidth available on the interface. Policy maps were then configured on the basis of that specific amount of bandwidth, meaning that separate policy maps were required for each interface.

This feature provides the ability to configure traffic policing and traffic shaping based on a *percentage* of bandwidth available on the interface. Configuring traffic policing and traffic shaping in this manner enables customers to use the same policy map for multiple interfaces with differing amounts of bandwidth.

This feature also provides the option of specifying burst sizes in milliseconds (ms) when configuring traffic policing and shaping based on a percentage of bandwidth.

## Benefits

### Increased Flexibility

This feature provides the ability to configure traffic policing and traffic shaping based on a *percentage* of bandwidth available on an interface. Configuring traffic policing and traffic shaping in this manner enables customers to use the same policy map for multiple interfaces with differing amounts of bandwidth.

## Restrictions

The **shape** (percent) command, when used in "child" (nested) policy maps, is not supported on the Cisco 7500, the Cisco 7200, or lower series routers. Therefore, the **shape** (percent) command cannot be configured for use in nested policy maps on these routers.

## Related Features and Technologies

- Modular QoS command-line interface (CLI) (Modular QoS CLI)
- Class-based weighted fair queueing (CBWFQ)
- Class-based packet marking
- Cisco Express Forwarding (CEF) and Distributed CEF (dCEF)
- Traffic policing
- Two-rate policing
- Traffic shaping

# Related Documents

- "Applying QoS Features Using the MQC" module
- "Configuring Weighted Fair Queueing" module
- "Marking Network Traffic" module
- "Policing and Shaping Overview" module
- "Traffic Policing" module
- "Two-Rate Policer" module
- "Policer Enhancements-Multiple Actions" module
- "Cisco Express Forwarding Overview" module
- Cisco IOS Quality of Service Solutions Command Reference
- Cisco IOS Switching Services Command Reference
- RFC 2697, *A Single Rate Three Color Marker*
- RFC 2698, *A Two Rate Three Color Marker*

# Supported Standards MIBs and RFCs

### Standards

None

### MIBs

None

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

### RFCs

- RFC 2697, *A Single Rate Three Color Marker*

- RFC 2698, *A Two Rate Three Color Marker*

# Prerequisites

On a Cisco 7500 series router, Distributed Cisco Express Forwarding (dCEF) must be configured on the interface before you can use the Percentage-Based Policing and Shaping feature.

# Configuration Tasks

## Configuring Policing and Shaping Based on Bandwidth Percentage

### SUMMARY STEPS

1. Router (config)# **policy-map***policy-name*
2. Router(config-pmap)# **class-map***class-map-name*
3. Router(config-pmap-c)# **police cir percent** *percent*[**bc***conform-burst-in-msec* ] [ **pir percent** *percent* ] [**be***peak-burst-in-msec* ]
4. Router(config-pmap-c)# **shape** {**average** | **peak**} **percent***percent* [*bc* ] [*be* ]
5. Router (config-pmap-c)# **service-policy***policy-map-name*
6. Router(config-pmap-c)# **exit**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Router (config)# **policy-map***policy-name* | Specifies the name of the policy map to be created. Enters policy-map configuration mode. |
| **Step 2** | Router(config-pmap)# **class-map***class-map-name* | Specifies the name of the class map to be created. Enters policy-map class configuration mode. |
| **Step 3** | Router(config-pmap-c)# **police cir percent** *percent*[**bc***conform-burst-in-msec* ] [ **pir percent** *percent* ] [**be***peak-burst-in-msec* ] | Configures traffic policing. |
| **Step 4** | Router(config-pmap-c)# **shape** {**average** \| **peak**} **percent***percent* [*bc* ] [*be* ] | Configures traffic shaping using either an average or peak traffic shaping rate based on a percentage of available bandwidth. |
| **Step 5** | Router (config-pmap-c)# **service-policy***policy-map-name* | Specifies the name of a policy map to be used as a child policy map for this class. |
| **Step 6** | Router(config-pmap-c)# **exit** | Exits policy-map class configuration mode. |

## Attaching the Policy Map to an Interface or a VC

| Command | Purpose |
|---|---|
| `Router(config-if)#` **service-policy output** *policy-map-name*<br>`Router(config-if-atm-vc)#`<br>**sevice-policy output** *policy-map-name* | Specifies the name of the policy map to be attached to the input direction of an interface or VC.<br>The policy map evaluates all traffic entering that interface or VC.<br>**Note** Traffic shaping is supported on service policies attached to output interfaces or output VCs only. |

## Verifying the Policing and Shaping Bandwidth Percentage Setting

| Command | Purpose |
|---|---|
| `Router#` **show class-map** | Displays all information about a class map, including the match criterion. |
| `Router#` **show policy-map** | Displays all configured policy maps. |
| `Router#` **show policy-map interface** *`interface-name`* | Displays the packet statistics of all classes that are configured for all service policies either on the specified interface or subinterface or on a specific permanent virtual circuit (PVC) on the interface. |

## Troubleshooting Tips

- For input traffic policing on a Cisco 7500 series router, verify that dCEF is enabled on the interface on which traffic policing is configured.
- For output traffic policing on a Cisco 7500 series router, ensure that the incoming traffic is dCEF-switched. Traffic policing cannot be used on the switching path unless dCEF switching is enabled.

# Configuration Examples

## Example Specifying Traffic Policing Based on a Bandwidth Percentage

The following example configures traffic policing using a committed information rate (CIR) and a peak information rate (PIR) based on a percentage of bandwidth. In this example, a CIR of 20 percent and a PIR of 40 percent have been specified. Additionally, an optional bc value and be value (300 ms and 400 ms, respectively) have been specified.

```
Router (config)#
 policy-map policy1

Router(config-pmap)# class-map class1

Router(config-pmap-c)# police cir percent 20 bc 300 ms pir percent 40 be 400 ms

Router (config-pmap-c)# service-policy child-policy1

Router(config-pmap-c)# exit


Router(config-pmap-c)# interface serial 3/1

Router(config-if)#
 service-policy output policy1
```

The purpose of the burst parameters (bc and be values) is to drop packets gradually, as is done with Weighted Random Early Detection (WRED), and to avoid tail drop. Setting sufficiently high burst values helps to ensure good throughput.

## Example Specifying Traffic Shaping Based on a Bandwidth Percentage

The following example configures traffic shaping using an average shaping rate based on a percentage of bandwidth. In this example, 25 percent of the bandwidth has been specified. Additionally, an optional bc value and be value (300 ms and 400 ms, respectively) have been specified.

```
Router (config)#
 policy-map policy1

Router(config-pmap)# class-map class1

Router(config-pmap-c)# shape average percent 25 300 ms 400 ms

Router (config-pmap-c)# service-policy child-policy1

Router(config-pmap-c)# exit


Router(config-pmap-c)# interface serial 3/1

Router(config-if)#
 service-policy output policy1
```

The purpose of the bc and be values is to drop packets gradually, as is done with WRED, and to avoid tail drop. Setting sufficiently high burst values helps to ensure good throughput.

## Example Verifying That CEF Is Enabled

On a Cisco 7500 series router, dCEF must be configured on the interface before you can use the Percentage-Based Policing and Shaping feature. The **show ip cef summary** command can be used to confirm that dCEF is enabled and is being used for IP switching. In rare instances, this command displays "IP Distributed

CEF without switching" in the command output. This indicates that dCEF is disabled. The following sample output of the **show ip cef summary** command indicates that dCEF is disabled:

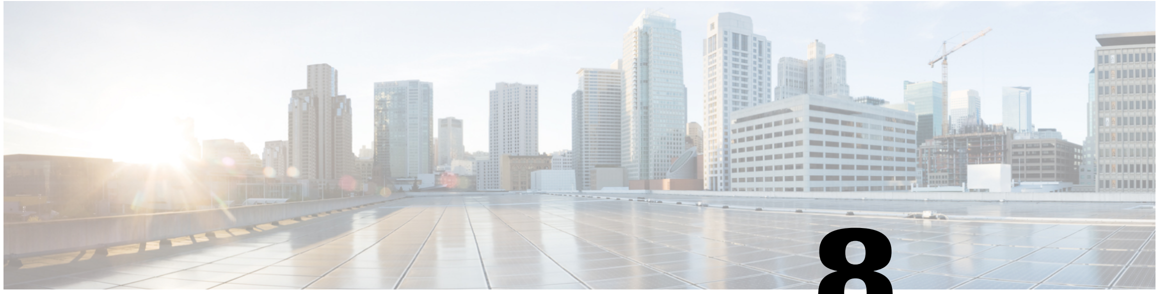
```
Router# show ip cef summary

    IP Distributed CEF with switching (Table Version 36), flags=0x0
      18 routes, 0 reresolve, 0 unresolved (0 old, 0 new), peak 3
      18 leaves, 19 nodes, 22136 bytes, 45 inserts, 27 invalidations
      0 load sharing elements, 0 bytes, 0 references
      universal per-destination load sharing algorithm, id 680E93E2
      3(0) CEF resets, 1 revisions of existing leaves
      Resolution Timer:Exponential (currently 1s, peak 1s)
      0 in-place/0 aborted modifications
      refcounts: 5136 leaf, 5120 node
```

When you configure a feature that requires special handling or is not yet supported in the dCEF switching paths, packets are forwarded to the next switching layer for handling. In this instance, the output of the **show cef interface** command displays "Packets switched to this interface on line card are dropped to next slow path" as shown in the following sample output.

```
Router# show cef interface Serial 10/0/0:28

    Serial10/0/0:28 is up (if_number 38)
      Internet address is 90.0.0.1/8
      ICMP redirects are never sent
      Per packet loadbalancing is disabled
      Inbound  access list is not set
      Interface is marked as point to point interface
      Packets switched to this interface on linecard are dropped to next slow path
Hardware idb is Serial10/0/0:28
      Fast switching type 4, interface type 20
      IP Distributed CEF switching enabled
      Fast flags 0x0. ifindex 37(37)
      Slot 10 Slot unit 0 VC 28
      Hardware transmit queue ptr 0x48001AE0 (0x48001AE0)
      Transmit limit accumulator 0x48000102 (0x48000102)
      IP MTU 1500
```



CHAPTER 8

# Modular QoS CLI Three-Level Hierarchical Policer

The Modular QoS CLI (MQC) Three-Level Hierarchical Policer extends the traffic policing functionality by allowing you to configure traffic policing at three levels of policy map hierarchies; a primary level, a secondary level, and a tertiary level. Traffic policing may be configured at any or all of these levels, depending on the needs of your network. Configuring traffic policing in a three-level hierarchical structure provides a high degree of granularity for traffic policing.

### Feature Specifications for the Modular QoS CLI (MQC) Three-Level Hierarchical Policer

| **Feature History** | |
|---|---|
| Release | Modification |
| 12.2(13)T | This feature was introduced. |
| Supported Platforms | |
| For platforms supported in Cisco IOS Release 12.2(13)T, consult Cisco Feature Navigator. | |

# 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 at the end of this module.

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

# Restrictions for the Modular QoS CLI Three-Level Hierarchical Policer

If traffic policing is configured at both the top level and secondary levels, note the following caveats:

- When traffic policing is configured at both the primary and secondary levels, the traffic policer at the secondary level acts only on packets sent by the policer at the top level.

However, the packet classification for the policy map at the secondary level occurs before the primary level policer has acted on the classes. When this situation occurs, the class counters for the policy map at the secondary level may not be equal to the number of packets acted upon by the second level policer.

The following output of the **show policy-map interface**command helps to illustrate this point. In this sample output two policy maps (called "primary_level," and "secondary_level," respectively) have been configured. The primary_level policy map contains a class map called "c1," and the secondary_level policy map contains a class map called "c3".

```
> > > show policy interface serial5/0.1
> > >  Serial5/0.1
> > >
> > >   Service-policy output: primary_level
> > >
> > >     Class-map: c1 (match-all)
> > >       24038 packets, 3004750 bytes
> > >       30 second offered rate 0 bps, drop rate 0 bps
> > >       Match: any
> > >       police:
> > >           cir 300000 bps, bc 9375 bytes
> > >         conformed 18105 packets, 2263125 bytes; actions:
> > >           transmit
> > >         exceeded 5933 packets, 741625 bytes; actions:             (*)
> > >           drop
> > >         conformed 0 bps, exceed 0 bps
> > >
> > >       Service-policy : secondary_level
> > >
> > >         Class-map: c3 (match-all)
> > >           24038 packets, 3004750 bytes

> > >           30 second offered rate 0 bps, drop rate 0 bps
> > >           Match: any
> > >           police:                (<= Indicates traffic policing has been configured)
> > >               cir 200000 bps, bc 3000 bytes
> > >               pir 250000 bps, be 3000 bytes
> > >             conformed 12047 packets, 1505875 bytes; actions:       (**)
> > >               set-frde-transmit
> > >             exceeded 3004 packets, 375500 bytes; actions:          (**)
> > >               set-frde-transmit
> > >             violated 3054 packets, 381750 bytes; actions:          (**)
> > >               set-frde-transmit
> > >             conformed 0 bps, exceed 0 bps, violate 0 bps
> > >
> > >         Class-map: class-default (match-any)
> > >           0 packets, 0 bytes
> > >           30 second offered rate 0 bps, drop rate 0 bps
> > >           Match: any
> > >             0 packets, 0 bytes
> > >             30 second rate 0 bps
```

Note the following about this example:

- - The class counter for the class map called "c3" shows 24038 packets (italicized in the example).
  - Traffic policing has been configured in the policy map, and the traffic policing feature for class map "c3" shows a total of 18105 packets -- 12047 conformed packets, plus 3004 exceeded packets, plus 3054 violated packets (indicated by the double asterisks ("**") in the example). This total is because 5933 packets have already been dropped in class map "c1" (indicated by the "*" in the example).
  - Therefore, only 18105 packets (24038 packets minus 5933 packets) are acted upon by the traffic policing feature configured in the second_level policy map.
- In this implementation of the Modular QoS CLI (MQC) Three-Level Hierarchical Policer, traffic policing at the primary level does not guarantee fairness in sharing bandwidth among the child classes. If packets from two different classes arrive at the same rate and then go through a traffic policer, the output rates of the two classes could be different because this feature acts as an aggregate policer.

In other words, it is possible that the primary-level policer could drop packets in one class in favor of the other class. This situation would happen because the primary-level policer had enough tokens when the packets for one class arrived, but there were not enough tokens left for the other class. This pattern could continue indefinitely, based on the arrival pattern of the packets.

# Information About the Modular QoS CLI Three-Level Hierarchical Policer

## Modular Quality of Service Command-Line Interface

The MQC is a command-line interface (CLI) structure that allows you to create traffic policies and attach these policies to interfaces.

In the MQC, the **class-map** command is used to define a traffic class (which is then associated with a traffic policy). The purpose of a traffic class is to classify traffic.

The Modular quality of service (QoS) CLI structure consists of the following three processes:

- Defining a traffic class with the **class-map** command.
- Creating a traffic policy by associating the traffic class with one or more QoS features (using the **policy-map** command).
- Attaching the traffic policy to the interface with the **service-policy** command.

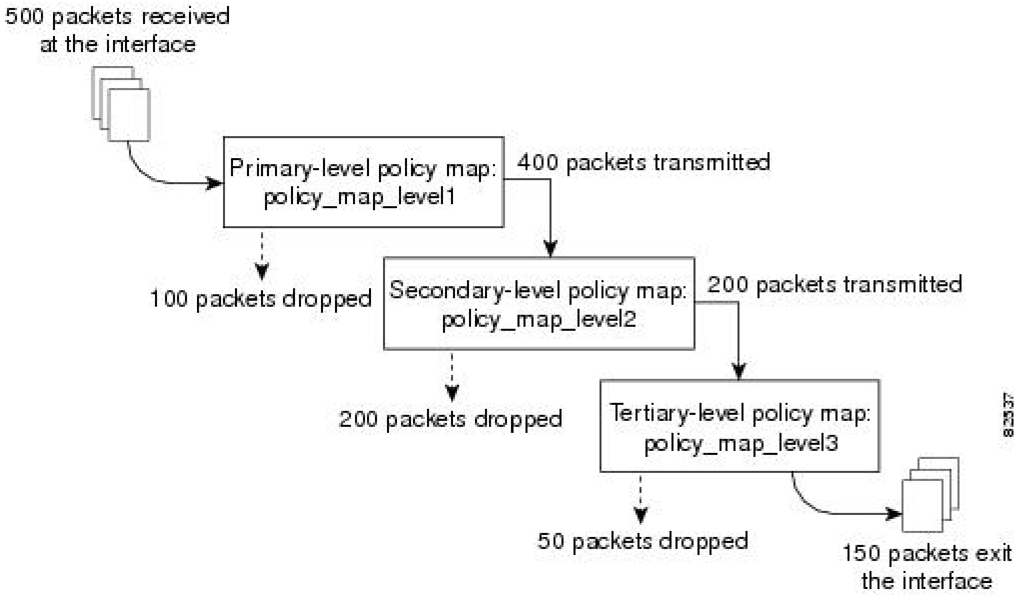
A traffic class contains three major elements: a name, a series of **match** commands, and, if more than one **match** command exists in the traffic class, an instruction on how to evaluate these **match** commands. The traffic class is named in the **class-map** command line; that is, if you enter the **class-map cisco**command while configuring the traffic class in the CLI, the traffic class would be named "cisco".

The **match** commands are used to specify various criteria for classifying packets. Packets are checked to determine whether they match the criteria specified in the **match** commands. If a packet matches the specified criteria, that packet is considered a member of the class and is forwarded according to the QoS specifications set in the traffic policy. Packets that fail to meet any of the matching criteria are classified as members of the default traffic class.

# Packet Flow in the Modular QoS CLI Three-Level Hierarchical Policer

The figure below illustrates the flow of packets among policy maps configured for traffic policing at each level in the hierarchy.

***Figure 1: Packet Flow Among Policy Maps***



In the figure above, three policy maps are configured: policy_map_level1 (the primary-level policy map), policy_map_level2 (the secondary-level policy map), and policy_map_level3 (the tertiary-level policy map). Traffic policing is configured in each policy map, and each policy map is attached to a service policy and to an interface.

In this simplified illustration, 500 packets arrive at the interface at which the policy map called "policy_map_level1" is attached. Because of the way traffic policing is configured in this policy map, 100 packets are dropped and 400 packets are transmitted.

The traffic policer at the secondary-level policy map (policy_map_level2) then evaluates the packets and treats them as determined by the way traffic policing is configured at this level. Of the 400 packets received, 200 are dropped and 200 are transmitted.

The traffic policer at the tertiary-level policy map (policy_map_level3), in turn, evaluates the 200 packets it has now received and applies the appropriate treatment as determined by the way the traffic policing is configured at this level.

## Other Traffic Policing-Related Features

The Cisco IOS traffic policing software features allow you to control the maximum rate of traffic sent or received on an interface. Traffic policing is often configured on interfaces at the edge of a network to limit traffic into or out of the network. Traffic that falls within the rate parameters is sent, whereas traffic that exceeds or violates the parameters is dropped or sent with a different priority.

The Cisco IOS software currently includes the following traffic policing features:

- Traffic Policing (a single-rate policer)
- Two-Rate Policer
- Policer Enhancements -- Multiple Actions
- Percentage-Based Policing and Shaping

Previously, these features could be configured at two levels of a policy map hierarchy; the top level and one secondary level. With the Modular QoS CLI (MQC) Three-Level Hierarchical Policer, these traffic policing-related features can configured in three levels of a policy map hierarchy.

The tasks for configuring each of these traffic policing-related features is essentially the same. That is, you use the MQC to create a policy map. Then you use the **police** command to configure traffic policing for a specific class within that policy map. The policy map is then attached to an interface.

Traffic policing can be configured to specify multiple marking actions for the traffic being policed, or to use a percentage of available bandwidth when policing traffic.

# How to Configure the Modular QoS CLI Three-Level Hierarchical Policer

## Configuring Traffic Policing

Traffic policing can be configured at any level of the policy map hierarchy, that is, at the primary level, secondary level, or the tertiary level.

### Before You Begin

Before configuring traffic policing, you must use the MQC to create a policy map.

## SUMMARY STEPS

1. **enable**
2. **configure** {**terminal** | **memory** | **network**}
3. **policy-map** *policy-name*
4. **class-map** *class-map-name*
5. **police** *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action* **violate-action** *action*
6. **exit**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables higher privilege levels, such as privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure** {**terminal** \| **memory** \| **network**}<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **policy-map** *policy-name*<br>**Example:**<br>`Router(config)# policy-map policy1` | Specifies the name of the policy map created earlier and enters policy-map configuration mode.<br>• See the Configuring Traffic Policing.<br>• Enter policy map name. |
| **Step 4** | **class-map** *class-map-name*<br>**Example:**<br>`Router(config-pmap)# class-map class1` | Specifies the name of the class map created when the policy map was created earlier and enters policy-map class configuration mode.<br>• See the Configuring Traffic Policing.<br>• Enter the class map name. |
| **Step 5** | **police** *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action* **violate-action** *action*<br>**Example:**<br>`Router(config-pmap-c)# police 8000 1000 1000 conform-action transmit exceed-action drop violate-action drop` | Configures traffic policing according to burst sizes and any optional actions specified. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | **exit**<br>**Example:**<br>`Router(config-pmap-c)# exit` | (Optional) Exits the policy-map class configuration mode. |

# Attaching the Policy Map to an Interface

After the policy map has been created and traffic policing has been configured, the policy map must be attached to an interface. Policy maps can be attached to either the input or output direction of the interface.

Depending on the needs of your network, you may need to attach the policy map to a subinterface, an ATM permanent virtual circuit (PVC), a Frame Relay data-link connection identifier (DLCI), or other type of interface.

### SUMMARY STEPS

1. **enable**
2. **configure** {**terminal** | **memory** | **network**}
3. **interface** *type number*
4. **pvc** [*name*] *vpi* / *vci* [**ilmi** | **qsaal** | **smds**]
5. **service-policy** {**input**| **output**} *policy-map-name*
6. **exit**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables higher privilege levels, such as privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure** {**terminal** \| **memory** \| **network**}<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **interface** *type number*<br>**Example:**<br>**Example:**<br>`Router(config-if)#`<br>`interface s4/0` | Configures an interface (or subinterface) type and enters interface configuration mode.<br>• Enter the interface type number. |
| **Step 4** | **pvc** [*name*] *vpi* / *vci* [**ilmi** \| **qsaal** \| **smds**]<br>**Example:**<br>`Router(config-if)# pvc cisco 0/16 ilmi` | (Optional) Creates or assigns a name to an ATM PVC and specifies the encapsulation type on an ATM PVC. Enters ATM virtual circuit (VC) configuration mode (config-if-atm-vc).<br>**Note** This step is required only if you are attaching the policy map to an ATM PVC. If you are not attaching the policy map to an ATM PVC, skip this step and proceed with Attaching the Policy Map to an Interface. |
| **Step 5** | **service-policy** {**input**\| **output**} *policy-map-name*<br>**Example:**<br>`Router(config-if)#`<br>`service-policy input policy1`<br>**Example:** | Specifies the name of the policy map to be attached to the input *or* output direction of the interface.<br>**Note** Policy maps can be configured on ingress or egress routers. They can also be attached in the input or output direction of an interface. The direction (input or output) and the router (ingress or egress) to which the policy map should be attached varies according your network configuration. When using the **service-policy** command to attach the policy map to an interface, be sure to choose the router and the interface direction that are appropriate for your network configuration.<br>• Enter the policy map name. |
| **Step 6** | **exit**<br>**Example:**<br>`Router(config-if)# exit` | (Optional) Exits interface configuration mode. |

## What to Do Next

If you want to configure traffic policing at another level in the policy map hierarchy, repeat the steps in the Configuring Traffic Policing, on page 55 section and the Attaching the Policy Map to an Interface, on page 57 section.

# Verifying the Configuration

This task allows you to verify that you created the configuration you intended and that the feature is functioning correctly.

### SUMMARY STEPS

1. **enable**
2. Do one of the following:
   - **show policy-map**
   - 
   - **show policy-map interface** *interface-name*
3. **exit**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables higher privilege levels, such as privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | Do one of the following:<br>• **show policy-map**<br>•<br>• **show policy-map interface** *interface-name*<br>**Example:**<br>`Router# show policy-map`<br>**Example:**<br>**Example:**<br>`Router#`<br>`show policy-map interface s4/0` | Displays all configured policy maps.<br>or<br>Displays the packet statistics of all classes that are configured for all service policies either on the specified interface or subinterface or on a specific PVC on the interface.<br>• Enter the interface name. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **exit**<br>**Example:**<br>`Router(config-if)# exit` | (Optional) Exits interface configuration mode. |

## Troubleshooting Tips

The commands in the Verifying the Configuration, on page 59 section allow you to verify that you achieved the intended configuration and that the feature is functioning correctly. If after using the **show** commands listed above, the configuration is not correct or the feature is not functioning as expected, do the following:

If the configuration is not the one you intended, complete the following procedures:

- Use the **show running-config** command and analyze the output of the command.
- If the policy map does not appear in the output of the **show running-config** command, enable the **logging console** command.
- Attach the policy map to the interface again.

If the packets are not being matched correctly (for example, the packet counters are not incrementing correctly), complete the following procedures:

- Use the **show policy-map**command and analyze the output of the command.
- Use the **show running-config** command and analyze the output of the command.
- Run the **show policy-map interface** command and analyze the output of the command. Review the the following:
  - If a policy map applies queueing, and the packets are matching the correct class, but you see unexpected results, compare the number of packets to the number of packets matched.
  - If the interface is congested, and you are only seeing a small number of packets matched, check the tuning of the tx ring, and evaluate whether the queueing is happening on the tx ring. To do this, use the **show controllers** command, and look at the value of the tx count in the show output of the command.

# Configuration Examples for the Modular QoS CLI Three-Level Hierarchical Policer

## Example Configuring the Modular QoS CLI Three-Level Hierarchical Policer

In the following example, the Modular QoS CLI (MQC) Three-Level Hierarchical Policer has been configured for three classes within three separate policy maps. The three classes, called "c1," "c2," and "c3," respectively, have been configured using the match criteria specified as follows:

```
class-map c1
   match any
class-map c2
   match ip precedence 1 2 3
class-map c3
   match ip precedence 2
```

Next, the classes are configured in three separate policy maps, called "p_all" (the primary-level policy map), "pmatch_123" (the secondary-level policy map), and "pmatch_2" (the tertiary-level policy map), as shown below.

```
policy p_all
   class c1
      police 100000
      service-policy pmatch_123
policy pmatch_123
  class c2
     police 20000
     service-policy pmatch_2
policy pmatch_2
  class c3
     police 8000
```

The primary goal of this configuration is to limit all traffic to 100 kbps. Within this, the secondary goal is make sure that packets with precedence values of 1, 2, or 3 do not exceed 20 kbps and that packets with precedence value of 2 never exceed 8 kbps.

To verify that the classes have been configured correctly and to confirm the results of the traffic policing configuration in the policy maps, the **show policy-map** command and the **show policy-map interface**command can be used, as shown in the following sections.

The following sample output of the **show policy-map**command verifies the configuration of the classes in the policy maps:

```
Router# show policy map
  Policy Map p_all
    Class c1
     police cir 100000 bc 3000
       conform-action transmit
       exceed-action drop
      service-policy pmatch_123
  Policy Map pmatch_123
    Class c2
     police cir 20000 bc 1500
       conform-action transmit
       exceed-action drop
      service-policy pmatch_2
  Policy Map pmatch_2
    Class c3
     police cir 8000 bc 1500
```

```
       conform-action transmit
       exceed-action drop
```

The following sample output of the **show policy-map interface** command confirms the results of this configuration on the attached interface:

```
Router# show policy-map interface Ethernet3/1
 Ethernet3/1
  Service-policy output:p_all
    Class-map:c1 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:any
      police:
          cir 100000 bps, bc 3000 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0 bps, exceed 0 bps,
      Service-policy :pmatch_123
        Class-map:c2 (match-all)
          0 packets, 0 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match:ip precedence 1  2  3
          police:
              cir 20000 bps, bc 1500 bytes
            conformed 0 packets, 0 bytes; actions:
              transmit
            exceeded 0 packets, 0 bytes; actions:
              drop
            conformed 0 bps, exceed 0 bps,
          Service-policy :pmatch_2
            Class-map:c3 (match-all)
              0 packets, 0 bytes
              5 minute offered rate 0 bps, drop rate 0 bps
              Match:ip precedence 2
              police:
                  cir 8000 bps, bc 1500 bytes
                conformed 0 packets, 0 bytes; actions:
                  transmit
                exceeded 0 packets, 0 bytes; actions:
                  drop
                conformed 0 bps, exceed 0 bps,
            Class-map:class-default (match-any)
              0 packets, 0 bytes
              5 minute offered rate 0 bps, drop rate 0 bps
              Match:any
        Class-map:class-default (match-any)
          0 packets, 0 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match:any
    Class-map:class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:any
```

# Additional References

The following sections provide additional references related to the Modular QoS CLI (MQC) Three-Level Hierarchical Policer:

### Related Documents

| Related Topic | Document Title |
|---|---|
| QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | *Cisco IOS Quality of Service Solutions Command Reference* |
| Additional information about configuring traffic policing | "Policing and Shaping Overview" module |
| Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC) | "Applying QoS Features Using the MQC" module |
| Two-rate traffic policing | "Two-Rate Policer" module |
| Traffic policing using multiple policer actions | "Policer Enhancements--Multiple Actions" module |
| Percentage-based traffic policing and shaping | "Percentage-Based Policing and Shaping" module |
| Frame Relay configurations | "Configuring Frame Relay" module |
| Frame Relay commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | *Cisco IOS Wide-Area Networking Command Reference* |

### Standards

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

### MIBs

| MIBs | MIBs Link |
|---|---|
| • CISCO-CLASS-BASED-QOS-CAPABILITY-MIB<br>• CISCO-CLASS-BASED-QOS-MIB | To obtain lists of supported MIBs by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB website on Cisco.com at the following URL:<br>http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml |

### RFCs

| RFCs | Title |
|---|---|
| RFC 2697 | *A Single Rate Three Color Marker* |
| RFC 2698 | *A Two Rate Three Color Marker* |

### Technical Assistance

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

CHAPTER 9

# ATM Policing by Service Category for SVC and SoftPVC

### Feature History

| Release | Modification |
|---|---|
| 12.2(4)B | This feature was introduced on the Cisco 6400 NSP. |
| 12.2(13)T | This command was integrated into Cisco IOS Release 12.2(13)T. |

This module describes the ATM Policing by Service Category for SVC/SoftPVC feature in Cisco IOS Release 12.2(13)T and includes the following sections:

# 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 at the end of this module.

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.

# Feature Overview

When configured, an ATM switch at the network side of a user-to-network (UNI) interface polices the flow of cells in the forward (into the network) direction of a virtual connection. These traffic policing mechanisms are known as usage parameter control (UPC). With UPC, the switch determines whether received cells comply with the negotiated traffic management values and takes one of the following actions on violating cells:

- Pass the cell without changing the cell loss priority (CLP) bit in the cell header.
- Tag the cell with a CLP bit value of 1.
- Drop (discard) the cell.

The SVC/SoftPVC feature enables you to specify which traffic to police, based on service category, on switched virtual circuits (SVCs) or terminating VCs on the destination end of a soft VC.

## Benefits

This feature enables you to select which and how traffic is affected by UPC. For example, you can configure your switch to pass all UBR traffic, but tag all other traffic types.

## Related Features and Technologies

- Intelligent early packet discard (EPD)
- Intelligent partial (tail) packet discard

## Related Documents

- ATM Switch Router Software Configuration Guide
- ATM and Layer 3 Switch Router Command Reference
- Guide to ATM Technology
- ATM Forum UNI 3.1 Specification

# Supported Platforms

This feature is supported on the node switch processor (NSP) of the Cisco 6400 carrier-class broadband aggregator.

# Supported Standards MIBs and RFCs

### Standards

None

### MIBs

CISCO-ATM-IF-MIB.my--New objects were created for per-service category SVC UPC intent.

To obtain lists of supported MIBs by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB website on Cisco.com at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

### RFCs

None

# Configuration Tasks

## Configuring ATM Policing by Service Category for SVC and SoftPVC

### SUMMARY STEPS

1. Switch(config)# **interface atm***slot***/***subslot***/***port*
2. Switch(config-if)# **atm svc-upc-intent** [{**abr** | **cbr** | **vbr-rt** | **vbr-nrt** | **ubr**}] {**tag** | **pass** | **drop**}

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Switch(config)# **interface atm***slot***/***subslot***/***port* | Selects the ATM interface. |
| **Step 2** | Switch(config-if)# **atm svc-upc-intent** [{**abr** \| **cbr** \| **vbr-rt** \| **vbr-nrt** \| **ubr**}] {**tag** \| **pass** \| **drop**}<br>**Example:**<br>**Example:**<br>`(Repeat this step for each service category and UPC mode combination.)` | Specifies the UPC mode. If no service category is specified, then the UPC mode configuration is applied to all traffic types. |

# Verifying ATM Policing by Service Category for SVC and SoftPVC

### SUMMARY STEPS

1. Enter the **show atm vc** or **show atm vp** EXEC command to display the UPC mode for a particular VC or VP.
2. Enter the **show atm interface** EXEC command. If the UPC mode is not the same for all service categories, the "Svc Upc Intent" field displays "by sc."

### DETAILED STEPS

**Step 1** Enter the **show atm vc** or **show atm vp** EXEC command to display the UPC mode for a particular VC or VP.

**Example:**

```
Switch# show atm vc int atm 0/0/1 2 120
Interface:ATM0/0/1, Type:oc3suni
VPI = 2  VCI = 120
Status:DOWN
Time-since-last-status-change:1w1d
Connection-type:PVC
Cast-type:point-to-multipoint-leaf
Packet-discard-option:disabled
Usage-Parameter-Control (UPC):pass

Wrr weight:2
Number of OAM-configured connections:0
OAM-configuration:disabled
OAM-states: Not-applicable
Cross-connect-interface:ATM0/0/1, Type:oc3suni
...
```

**Step 2** Enter the **show atm interface** EXEC command. If the UPC mode is not the same for all service categories, the "Svc Upc Intent" field displays "by sc."

**Example:**

```
Switch# show atm interface atm 8/0/1
Interface:     ATM8/0/1        Port-type:     oc3suni
IF Status:     UP              Admin Status:  up
Auto-config:   enabled         AutoCfgState:  completed
IF-Side:       Network         IF-type:       NNI
Uni-type:      not applicable  Uni-version:   not applicable
Max-VPI-bits:  8               Max-VCI-bits:  14
Max-VP:        255             Max-VC:        16383
ConfMaxSvpcVpi:255             CurrMaxSvpcVpi:255
ConfMaxSvccVpi:255             CurrMaxSvccVpi:255
ConfMinSvccVci:35              CurrMinSvccVci:35
Svc Upc Intent:by sc
           Signalling:    Enabled
ATM Address for Soft VC:47.0091.8100.0000.0002.b9ae.9301.4000.0c84.0010.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      3        4       0       0       1        0       0          8          7
Logical ports(VP-tunnels):    0
Input cells:   3036674         Output cells:  3036816
5 minute input rate:            0 bits/sec,       0 cells/sec
```

```
5 minute output rate:           0 bits/sec,       0 cells/sec
Input AAL5 pkts:1982638, Output AAL5 pkts:1982687, AAL5 crc errors:0
```

## Troubleshooting Tips

If a VC is not configured with the appropriate UPC mode, make sure that the VC was set up after the **atm svc-upc-intent** command was configured. Changes to the UPC mode take affect after the VC is torn down and set up again.

# Monitoring and Maintaining ATM Policing by Service Category for SVC and SoftPVC

| Command | Purpose |
|---|---|
| `Switch#` **show atm interface** | Displays ATM-specific information about an ATM interface. |
| `Switch#` **show controllers atm** *slot/subslot/port* | Displays information about a physical port device. Includes dropped (or discarded) cells. |
| `Switch#` **show atm vc** [**interface atm** *slot/subslot/port* ] | Displays the configured UPC action and intelligent packet discard mechanisms, as well as the number of cells discarded due to UPC violations. |

## Example Monitoring and Maintaining ATM Policing by Service Category for SVC and SoftPVC

```
Switch# show atm vc interface atm 3/0/1.51 51 16

     Interface: ATM3/0/1.51, Type: oc3suni
     VPI = 51  VCI = 16
     Status: DOWN
     Time-since-last-status-change: 2w0d
     Connection-type: PVC
     Cast-type: point-to-point
     Packet-discard-option: enabled

     Usage-Parameter-Control (UPC): pass

     Wrr weight: 32
     Number of OAM-configured connections: 0
     OAM-configuration: disabled
     OAM-states:  Not-applicable
     Cross-connect-interface: ATM2/0/0, Type: ATM Swi/Proc
     Cross-connect-VPI = 0
```

```
Cross-connect-VCI = 73
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Encapsulation: AAL5ILMI
Threshold Group: 6, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0

Rx pkts:0, Rx pkt drops:0
Rx connection-traffic-table-index: 6
Rx service-category: UBR (Unspecified Bit Rate)
Rx pcr-clp01: 424
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 6
Tx service-category: UBR (Unspecified Bit Rate)
Tx pcr-clp01: 424
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
No AAL5 connection registered
```

# Configuration Examples

## Example Non-UBR Traffic Policing

In the following example, the UBR traffic on ATM 3/0/0 is passed while all other traffic is policed:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm svc-upc-intent ubr pass
Switch(config-if)# atm svc-upc-intent cbr tag
Switch(config-if)# atm svc-upc-intent vbr-rt tag
Switch(config-if)# atm svc-upc-intent vbr-nrt tag
Switch(config-if)# atm svc-upc-intent abr drop
```

CHAPTER 10

# Modular QoS CLI Unconditional Packet Discard

### Feature History

| Release | Modification |
|---|---|
| 12.2(13)T | This feature was introduced. |
| Supported Platforms | |
| For platforms supported in Cisco IOS Release 12.2(13)T, consult Cisco Feature Navigator. | |

This module describes the Modular QoS CLI (MQC) Unconditional Packet Discard feature in Cisco IOS Release 12.2(13)T. It includes the following sections:

## 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 at the end of this module.

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.

# Feature Overview

The Modular QoS CLI (MQC) Unconditional Packet Discard feature allows customers to classify traffic matching certain criteria and then configure the system to unconditionally discard any packets matching that criteria. The Modular QoS CLI (MQC) Unconditional Packet Discard feature is configured using the Modular Quality of Service Command-Line Interface (MQC) feature. Packets are unconditionally discarded by using the new **drop** command within the MQC.

## Benefits

### Enhanced System Utilization

This feature allows you to discard (drop), without any further system processing, the packets of a particular class. This function is very useful when you want to discard all the packets for nonessential applications (for instance, Internet browsing applications or unauthorized video applications) and allocate system resources to more essential applications. This feature allows the user to discard those nonessential packets and simultaneously obtain the bit and drop rate statistics for that particular class and the traffic within that class. The statistics are gathered through the CISCO-CLASS-BASED-QOS-MIB.

## Restrictions

Packets are unconditionally discarded by configuring the drop action inside a traffic class (inside of a policy map). This drop action is accomplished with the new **drop** command. Note the following restrictions for configuring the drop action within a traffic class:

- The discarding action is the only action that can be configured in a traffic class. That is, no other actions can be configured in the traffic class.
- When a traffic class is configured with the **drop** command, a "child" (nested) policy cannot be configured for this specific traffic class through the **service policy** command.
- The discarding action cannot be configured for the default class known as the class-default class.

## Related Features and Technologies

- Modular quality of service command-line interface (MQC)

## Related Documents

- "Applying QoS Features Using the MQC" module
- "Classifying Network Traffic" module
- "Marking Network Traffic" module
- Cisco IOS Quality of Service Solutions Command Reference

# Supported Standards MIBs and RFCs

### Standards

None

### MIBs

- CISCO-CLASS-BASED-QOS-MIB
- CISCO-CLASS-BASED-QOS-CAPABILITY-MIB

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

### RFCs

None

# Configuration Tasks

## Configuring the Class Map

### SUMMARY STEPS

1. Router(config)# **class-map***class-map-name*
2. Router(config-cmap)# **match access-group** {*access-group* | **name***access-group-name* }
3. Router(config-cmap)# **exit**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Router(config)# **class-map***class-map-name* | Specifies the name of the class map to be created. If match-all or match-any is not specified, traffic must match all the match criteria to be classified as part of the traffic class. Enters class-map configuration mode. |
| **Step 2** | Router(config-cmap)# **match access-group** {*access-group* \| **name***access-group-name* } | Specifies that traffic matching the specified access group will be placed in the map class created above. This command provides just an example of the match criterion you can specify. For more information about the additional match criteria available, see the "Applying QoS Features Using the MQC" module. |
| **Step 3** | Router(config-cmap)# **exit** | Exits from the configuration mode. |

## Creating a Policy Map

### SUMMARY STEPS

1. Router (config)# **policy-map***policy-name*
2. Router (config-pmap)# **class***class-name*
3. Router (config-pmap)# **drop**
4. Router(config-cmap)# **exit**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Router (config)# **policy-map***policy-name*<br>**Example:** | Specifies the name of the policy map to be created. Enters policy-map configuration mode. |
| **Step 2** | Router (config-pmap)# **class***class-name* | Specifies the name of the traffic class configured earlier in the Configuring the Class Map, on page 73 section above. This traffic class is used to classify traffic to the policy map. Enters policy-map class configuration mode. |
| **Step 3** | Router (config-pmap)# **drop** | Discards the packets in the specified traffic class. |
| **Step 4** | Router(config-cmap)# **exit** | Exits policy-map configuration mode. |

# Attaching the Policy Map to an Interface or a VC

## SUMMARY STEPS

1. Router(config)# **interface***type number* [ *name-tag*
2. Router(config-if)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal** | **smds**]
3. Do one of the following:
   - Router(config-if)# **service-policy input***policy-map-name*
   -
   -
   -
4. Router(config-if)# **exit**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Router(config)# **interface***type number* [ *name-tag*<br>**Example:** | Configures the interface type and enters interface configuration mode. |
| **Step 2** | Router(config-if)# **pvc** [*name*] *vpi*/*vci* [**ilmi** \| **qsaal** \| **smds**] | (Optional) Creates or assigns a name to an ATM permanent virtual circuit (PVC), and specifies the encapsulation type on an ATM PVC.<br>Enters ATM VC configuration mode (config-if-atm-vc).<br>This step is required only if you are attaching the policy map to an ATM PVC. |
| **Step 3** | Do one of the following:<br>• Router(config-if)# **service-policy input***policy-map-name*<br>•<br>•<br>•<br>**Example:**<br>`Router(config-if-atm-vc)#` **service-policy output***`policy-map-name`*<br>**Example:** | Specifies the name of the policy map to be attached to the input or output direction of an interface or VC. The policy map evaluates all traffic entering or leaving that interface or VC. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | Router(config-if)# **exit** | Exits interface configuration mode. |

## Verifying the Discard Action Configuration in the Traffic Class

| Command | Purpose |
|---|---|
| `Router#` **show policy-map** | Displays the configuration of all classes for a specified service policy map or all classes for all existing policy maps. |
| `Router#` **show policy-map interface** *`interface-name`* | Displays the packet statistics of all classes that are configured for all service policies either on the specified interface or subinterface or on a specific PVC on the interface. |

# Configuration Examples

## Example Configuring the Discard Action Configuration in a Traffic Class

In the following sample configuration, a traffic class called "class1" has been created and configured for use in a policy-map called "policy1." The policy-map policy1 is attached to an output serial interface 2/0. All packets matching access-group 101 are placed in a class called "c1." Packets belonging to this class are discarded.

```
Router(config)# class-map class1
Router(config-cmap)# match access-group 101
Router(config-cmap)# policy-map policy1
Router(config-pmap)# class c1
Router(config-pmap-c)# drop
Router(config-pmap-c)# interface s2/0
Router(config-if)# service-policy output policy1
Router(config-if)# exit
```

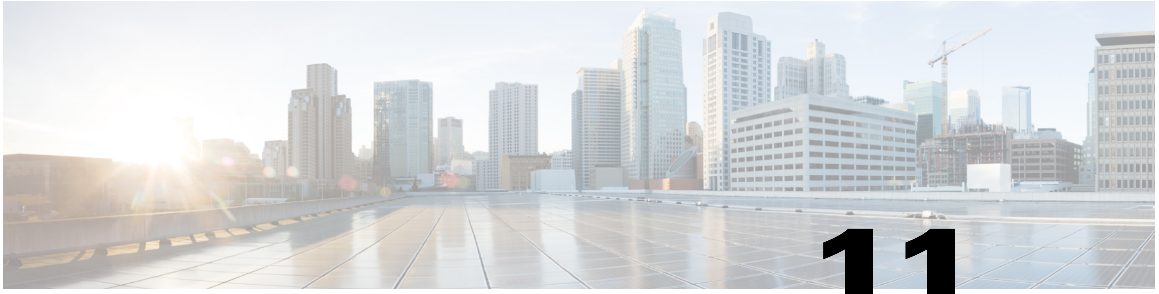
The following sample output of the **show policy-map** command displays the contents of the policy map called "policy1." All the packets belonging to the class called "c1" are discarded.

```
Router# show policy-map policy1
 Policy Map policy1
  Class c1
   drop
```

# Example Verifying the Discard Action Configuration in the Policy Map

The following sample output of the **show policy-map interface** command displays the statistics for the Serial2/0 interface, to which a policy map called "policy1" is attached. The discard action has been specified for all the packets belonging to a class called "c1." In this example, 32000 bps of traffic is sent ("offered") to the class and all of them are dropped. Therefore, the drop rate shows 32000 bps.

```
Router# show policy-map interface
 Serial2/0
 Serial2/0
  Service-policy output: policy1
    Class-map: c1 (match-all)
       10184 packets, 1056436 bytes
       5 minute offered rate 32000 bps, drop rate 32000 bps
       Match: ip precedence 0
       drop
```



CHAPTER 11

# Control Plane Policing

The Control Plane Policing feature allows you to configure a quality of service (QoS) filter that manages the traffic flow of control plane packets to protect the control plane of Cisco IOS routers and switches against reconnaissance and denial-of-service (DoS) attacks. In this way, the control plane (CP) can help maintain packet forwarding and protocol states despite an attack or heavy traffic load on the router or switch.

## 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 at the end of this module.

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

## Prerequisites for Control Plane Policing

The Modular Quality of Service (QoS) Command-Line interface (CLI) (MQC) is used to configure the packet classification and policing functionality of the Control Plane Policing feature.

Before configuring Control Plane Policing (CoPP), you should understand the procedures for using the MQC. For information about the MQC, see the "Applying QoS Features Using the MQC" module.

# Restrictions for Control Plane Policing

### Aggregate and Distributed Control Plane Policing

Aggregate policing is supported in Cisco IOS Release 12.0(29)S, Cisco IOS Release 12.2(18)S, Cisco IOS Release 12.3(4)T, and later releases.

Distributed policing is supported only in Cisco IOS Release 12.0(30)S and later Cisco IOS 12.0S releases.

### Output Rate-Limiting Support

Output rate-limiting is performed in silent (packet discard) mode. Silent mode enables a router to silently discard packets using policy maps applied to output control plane traffic with the **service-policy output** command. For more information, see the Output Rate-Limiting and Silent Mode Operation, on page 87.

Output rate-limiting (policing) in silent mode is supported only in:

- Cisco IOS Release 12.2(25)S and later Cisco IOS 12.2S releases
- Cisco IOS Release 12.3(4)T and later Cisco IOS 12.3T releases

Output rate-limiting is not supported for distributed control plane services in Cisco IOS 12.0S releases or in Cisco IOS 12.2SX releases.

Output rate-limiting is not supported on the Cisco 7500 series and Cisco 10720 Internet router.

### MQC Restrictions

The Control Plane Policing feature requires the MQC to configure packet classification and policing. All restrictions that apply when you use the MQC to configure policing also apply when you configure control plane policing. Only two MQC actions are supported in policy maps--**police** and **drop**.

**Note** On the Cisco 10720 Internet router, only the **police**command, not the **drop** command, is supported in policy maps. In addition, in a QoS service policy that is attached to the Cisco 10720 control plane, the **police**command does not support **set** actions as arguments in **conform-action**, **exceed-action**, and **violate-action** parameters.

Features that require Network-Based Application Recognition (NBAR) classification may not work well at the control plane level. The following classification (match) criteria are supported on all platforms:

- Standard and extended IP access lists (ACLs).
- In class-map configuration mode: **match ip dscp**, **match ip precedence**, and **match protocol arp,**and **match protocol pppoe**commands.

**Note** In the Cisco IOS 12.2SX release, the **match protocol arp** command is not supported.

On the Cisco 10720 Internet router, the following MQC commands are also supported in class-map configuration mode: **match input-interface**, **match mpls experimental**, **match protocol ipv6**, and **match qos-group**.

When using these commands for control plane policing on the Cisco 10720 Internet router, note the following restrictions:

- Packet classification using match criteria is not supported for packets that cannot be classified in the Cisco 10720 data path, such as unknown Layer 2 encapsulation and IP options.
- The following IPv6 fields are not be supported in packet classification for IPv6 QoS on the Cisco 10720 Internet router and are, therefore, not supported for control plane policing:
    - IPv6 source and destination addresses
    - Layer 2 class of service (CoS)
    - IPv6 routing header flag
    - IPv6 undetermined transport flag
    - IPv6 flow label
    - IP Real-Time transport Protocol (RTP)

**Note** Packets that are not supported for QoS packet classification on the Cisco 10720 Internet router are not policed in the default traffic class for control plane policing.

### CISCO-CLASS-BASED-QOS-MIB Control Plane Support

In Cisco IOS Release 12.3(7)T and later Cisco IOS 12.3T releases, the CISCO-CLASS-BASED-QOS-MIB is extended to manage control plane QoS policies and provide information about the control plane.

### Cisco IOS Release 12.2(18)SXD1

In Cisco IOS Release 12.2(18)SXD1 and later releases, Hardware Control Plane Interface for Control Plane Policing has the following restrictions:

- Supported only with Supervisor Engine 720. Not supported with Supervisor Engine 2.
- Does not support CoPP output rate-limiting (policing).
- Does not support the CoPP silent operation mode.
- Cisco IOS Release 12.2(18)SXD1 and later releases automatically install the CoPP service policy on all DFC-equipped switching modules.

For more information about control plane policing in Cisco IOS Release 12.2(18)SXD1 and later releases, see either of these publications:

- For Catalyst 6500 series switches, see the "Configuring Control Plane Policing (CoPP)" module.
- For Cisco 7600 series routers, see the "Configuring Denial of Service Protection" module.

# Information About Control Plane Policing

## Benefits of Control Plane Policing

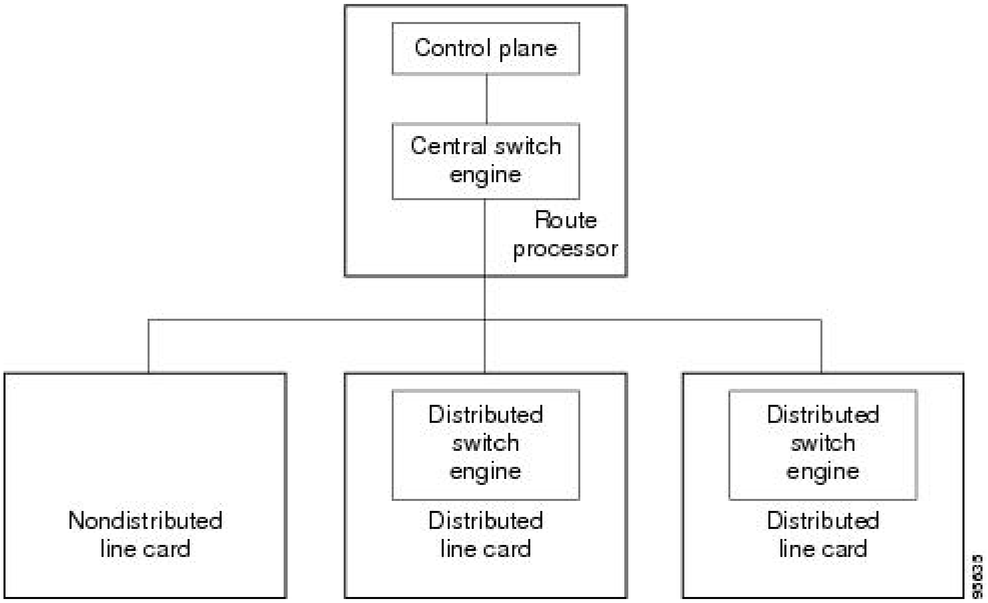
Configuring the Control Plane Policing feature on your Cisco router or switch provides the following benefits:

- Protection against DoS attacks at infrastructure routers and switches
- QoS control for packets that are destined to the control plane of Cisco routers or switches
- Ease of configuration for control plane policies
- Better platform reliability and availability

## Terms to Understand

Because different platforms can have different architectures, the following set of terms is defined. The figure below illustrates how control plane policing works.

*Figure 2: Layout of Control Plane, Central Switch Engine, Distributed Switch Engines, and Line Cards on a Router*



- Control plane (CP)--A collection of processes that run at the process level on the route processor (RP). These processes collectively provide high-level control for most Cisco IOS functions.
- Central switch engine--A device that is responsible for high-speed routing of IP packets. It also typically performs high-speed input and output services for nondistributed interfaces. (See nondistributed line cards.) The central switch engine is used to implement aggregate CP protection for all interfaces on the router.

**Note** All IP packets that are destined for the CP should pass through the central switch engine before they are forwarded to the process level.

On the Cisco 10720 Internet router, control plane policing is implemented on Cisco Parallel eXpress Forwarding (PXF) in a Toaster-based architecture. PXF is a hardware-based central switch engine that can filter traffic at a higher rate than the route processor. PXF switches all data traffic separately from the route processor. PXF packet processing occurs at an intermediate step between the nondistributed line cards and the route processor shown in the figure above. In addition to the regular punting, PXF also punts certain types of packets (such as unknown Layer 2 encapsulation and packets with IP options) to the RP for further processing at interrupt level.

**Note** On the Cisco 10720 Internet router, you can configure enhanced RP protection by using the **ip option drop** command to drop IPv4 packets with IP options that are punted to the RP by PXF. Tunneled IPv4 packets and IPv4 packets with an unsupported encapsulation method are not dropped. For more information, see the "ACL IP Options Selective Drop" module.

- Distributed switch engine--A device that is responsible for high-speed switching of IP packets on distributed line cards without using resources from the central switch engine. It also typically performs input and output services for the line card. Each distributed switch engine is used to implement distributed CP services for all ports on a line card. Input CP services distribute the processing load across multiple line cards and conserve vital central switch engine resources. Distributed CP services are optional; however, they provide a more refined level of service than aggregate services.
- Nondistributed line cards--Line cards that are responsible for receiving packets and occasionally performing input and output services. All packets must be forwarded to the central switch engine for a routing or switching decision. Aggregate CP services provide coverage for nondistributed line cards.

**Note** Distributed CP services are supported only in Cisco IOS Release 12.0(30)S and later 12.0S releases.

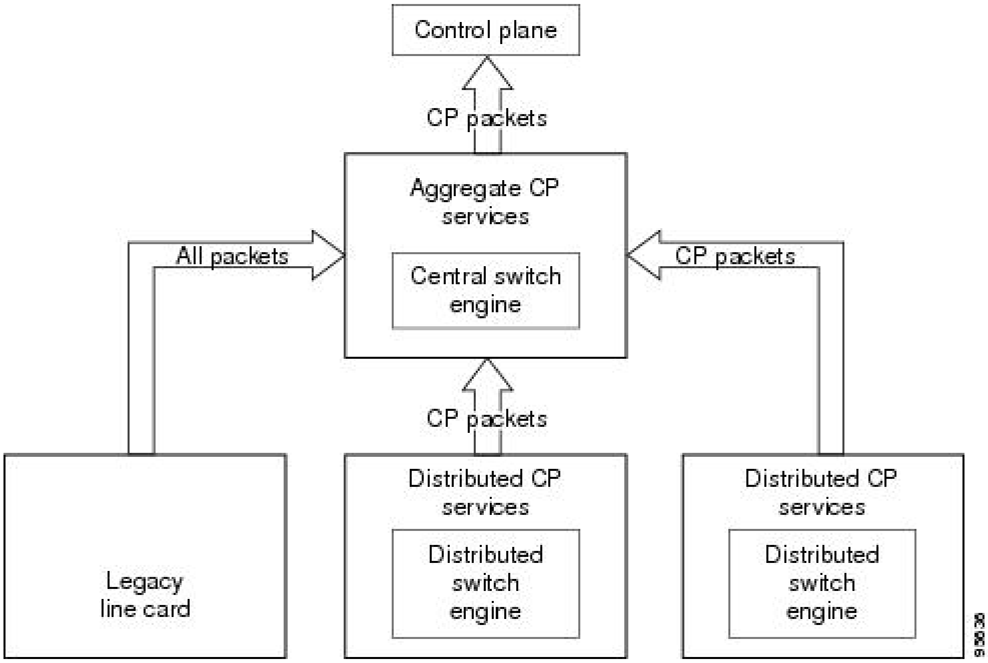
# Control Plane Security and Packet QoS Overview

To protect the CP on a router from DoS attacks and to provide packet QoS, the Control Plane Policing feature treats the CP as a separate entity with its own ingress (input) and egress (output) ports, which are like ports on a router and switch. Because the Control Plane Policing feature treats the CP as a separate entity, a set of rules can be established and associated with the ingress and egress ports of the CP.

These rules are applied only after the packet has been determined to have the CP as its destination or when a packet exits the CP. Thereafter, you can configure a service policy to prevent unwanted packets from progressing after a specified rate limit has been reached; for example, a system administrator can limit all TCP/SYN packets that are destined for the CP to a maximum rate of 1 megabit per second.

Input CP services are executed after router input port services have been performed and after a routing decision on the input path has been made. As shown in the figure below, CP security and packet QoS are applied on:

*Figure 3: Input Control Plane Services: Aggregate and Distributed Services*



The following types of Layer 3 packets are forwarded to the control plane and processed by aggregate and distributed control plane policing:

- Routing protocol control packets
- Packets destined for the local IP address of the router
- Packets from management protocols (such as Simple Network Management Protocol [SNMP], Telnet, and secure shell [SSH])

**Note** Ensure that Layer 3 control packets have priority over other packet types that are destined for the control plane.

# Aggregate Control Plane Services

Aggregate control plane services provide control plane policing for all CP packets that are received from all line-card interfaces on the router.

The central switch engine executes normal input port services and makes routing decisions for an incoming packet: if the packet is destined for the CP, aggregate services are performed. Because CP traffic from all line cards must pass through aggregate CP services, these services manage the cumulative amount of CP traffic that reaches the CP.

Aggregate CP service steps are as follows:

1 The line card receives a packet and delivers it to the central switch engine.

**Note** Before the packet is sent to the central switch engine, additional processing may be necessary for platforms that support hardware-level policing or platform-specific aggregate policing. It is possible that the packet may undergo multiple checks before it undergoes the generic Cisco IOS check.

1 The interfaces perform normal (interface-level) input port services and QoS.

2 The central switch engine performs Layer 3 switching or makes a routing decision, determining whether or not the packet is destined for the CP.

3 The central switch engine performs aggregate CP services for all CP packets.

4 On the basis of the results of the aggregate CP services, the central switch engine either drops the packet or delivers the packet to the CP for final processing.

### Functionality Highlights of Aggregate CP Services

The following list highlights the functionality of aggregate CP services:

- Aggregate CP services are defined for a single input interface, such as the CP, and represent an aggregate for all ports on a router.
- Modular QoS is used to define CP services. Class maps and policy maps for both DoS protection and packet QoS are defined for a single aggregate CP service policy.
- Modular QoS does not prevent a single bad port from consuming all allocated bandwidth. Class maps that match an interface or subinterface may be able to constrain the contribution of each interface through an interface-specific policy map.

## Distributed Control Plane Services

Distributed control plane services provide control plane policing for all CP packets that are received from the interfaces on a line card.

A distributed switch engine executes normal input port services and makes routing decisions for a packet: if the packet is destined for the CP, distributed CP services are performed. Afterwards, CP traffic from each line card is forwarded to the central switch engine where aggregate CP services are applied.

**Note** Distributed CP services may also forward conditioned packets to the central switch engine. In this case, aggregate CP services are also performed on the conditioned CP traffic.

Distributed CP service steps are as follows:

1 A line card receives a packet and delivers it to the distributed switch engine.

2 The distributed switch engine performs normal (interface-level) input port services and QoS.

3 The distributed switch engine performs Layer 2 or Layer 3 switching or makes a routing decision, determining whether the packet is destined for the CP.

4 The distributed switch engine performs distributed CP services for all CP packets.

5 On the basis of the results of the distributed CP services, the distributed switch engine either drops the packet or marks the packet and delivers it to the central switch engine for further processing.

6 The central switch engine performs aggregate CP services and delivers the packet to the CP for final processing.

### Functionality Highlights of Distributed CP Services

The following list highlights the functionality of distributed CP services:

- Distributed CP services are defined for a single input interface, such as the distributed CP, and represent an aggregate for all ports on a line card.
- The MQC is used to define CP services. Class maps and policy maps for both DoS protection and packet QoS are defined for a single distributed CP service policy. Each line card may have a unique CP service policy that applies traffic classifications, QoS policies, and DoS services to packets received from all ports on the line card in an aggregate way.
- The MQC does not prevent one bad port from consuming all allocated bandwidth on a line card. Class maps that match an interface or subinterface may be able to constrain the contribution of each interface through an interface-specific policy map.
- Distributed CP services allow you to limit the number of CP packets forwarded from a line card to the central switch engine. The total number of CP packets received from all line cards on a router may exceed aggregate CP levels.

## Usage of Distributed CP Services

The purpose of CP protection and packet QoS is to apply sufficient control to the packets that reach the control plane. To successfully configure this level of CP protection, you must:

- Apply traditional QoS services using the MQC to CP packets.
- Protect the path to the control plane against indiscriminate packet dropping due to resource exhaustion. If packets are not dropped according to user-defined QoS policies, but are dropped due to a resource limitation, the QoS policy is not maintained.

Distributed CP services allow you to configure specific CP services that are enforced at the line-card level and are required for the following reasons:

- While under a DoS attack, line-card resources may be consumed. In this case, you must configure a drop policy to identify important packets. The drop policy ensures that all important packets arrive to the central switch engine for aggregate CP protection and arrive later to the CP. Distributed CP services allow routers to apply the appropriate drop policy when resources are consumed and therefore maintain the desired QoS priorities. If a line card indiscriminately drops packets, the aggregate CP filter becomes ineffective and the QoS priorities are no longer maintained.
- It is not possible to prevent one interface from consuming all aggregate CP resources. A DoS attack on one port may negatively impact CP processing of traffic from other ports. Distributed CP services allow you to limit the amount of important traffic that is forwarded by a line card to the CP. For example, you

can configure a layered approach in which the combined rates of all line cards are over-subscribed compared to the aggregate rate. The rate of each individual line card would be below the aggregate rate, but combined together, the rates of all line cards exceed it. This over-subscription model is commonly used for other resource-related functions and helps limit the contribution of CP packets from any one line card.

- Distributed CP services provide for slot-level (line-card) filtering. Customer-facing interfaces may have greater security requirements (with more restrictions or for billing reasons) than network-facing interfaces to backbone devices.
- Because distributed CP protection allows you to configure packet filters on a per-line-card basis, processing cycles on line cards may offload aggregate level processing. You can configure Border Gateway Protocol (BGP) filtering at the distributed level for interfaces that use BGP, allowing the aggregate level to filter packets with the remaining filter requirements. Or you can configure identical filters for distributed and aggregate CP services with a distributed packet marking scheme that informs the aggregate filter that a packet has already been checked. Distributed CP service processing further reduces aggregate processing and can significantly reduce the load on aggregate CP services.

## Output Rate-Limiting and Silent Mode Operation

A router is automatically enabled to silently discard packets when you configure output policing on control plane traffic using the **service-policy output** *policy-map-name* command.

Rate-limiting (policing) of output traffic from the CP is performed in silent mode. In silent mode, a router that is running Cisco IOS software operates without sending any system messages. If a packet that is exiting the control plane is discarded for output policing, you do not receive an error message.

When control plane policing is configured for output traffic, error messages are not generated in the following cases:

- Traffic that is being transmitted to a port to which the router is not listening
- A connection to a legitimate address and port that is rejected because of a malformed request

The silent mode functionality and output policing on CP traffic are supported only in:

- Cisco IOS Release 12.2(25)S and later Cisco IOS 12.2S releases
- Cisco IOS Release 12.3(4)T and later Cisco IOS 12.3T releases

Silent mode and output policing on CP traffic are not supported for distributed control plane services.

# How to Use Control Plane Policing

## Defining Aggregate Control Plane Services

To configure aggregate CP services, such as packet rate control and silent packet discard, for the active route processor, complete the following steps.

### Before You Begin

Before you enter control-plane configuration mode to attach an existing QoS policy to the control plane, you must first create the policy using MQC to define a class map and policy map for control plane traffic.

**Note**

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control plane interface.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)

>

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane**
4. **service-policy** {**input**| **output** *policy-map-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. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **control-plane**<br>**Example:**<br>`Router(config)# control-plane` | Enters control-plane configuration mode (a prerequisite for Defining Aggregate Control Plane Services). |
| **Step 4** | **service-policy** {**input**\| **output** *policy-map-name* | Attaches a QoS service policy to the control plane. Note the following points: |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config-cp)# service-policy input control-plane-policy` | • **input** --Applies the specified service policy to packets received on the control plane.<br>• **output** --Applies the specified service policy to packets transmitted from the control plane and enables the router to silently discard packets.<br>• *policy-map-name* --Name of a service policy map (created using 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-cp)# end` | (Optional) Returns to privileged EXEC mode. |

# Defining Distributed Control Plane Services

To configure distributed CP services, such as packet rate control, for packets that are destined for the CP and sent from the interfaces on a line card, complete the following steps.

### Before You Begin

Before you enter control-plane configuration mode to attach an existing QoS policy for performing distributed control-plane services, you must first create the policy using MQC to define a class map and policy map for control-plane traffic.

**Note**

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control plane interface.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)
- With Cisco IOS 12.2SX releases, the Supervisor Engine 720 automatically installs the service policy on all DFC-equipped switching modules.

>

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**slot** *slot-number*]
4. **service-policy input** *policy-map-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. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **control-plane** [**slot** *slot-number*]<br>**Example:**<br>`Router(config)# control-plane slot 3` | Enters control-plane configuration mode, which allows you to optionally attach a QoS policy (used to manage CP traffic) to the specified slot.<br>• Enter the **slot** keyword and the slot number, as applicable. |
| **Step 4** | **service-policy input** *policy-map-name*<br>**Example:**<br>`Router(config-cp)# service-policy input control-plane-policy` | Attaches a QoS policy map to filter and manage CP traffic on a specified line card before the aggregate CP policy is applied. Note the following points:<br>• **input** --Applies the specified policy map using the distributed switch engine to CP packets that are received from all interfaces on the line card.<br>• *policy-map-name* --Name of a service policy map (created using the **policy-map** command) to be attached. The name can be a maximum of 40 alphanumeric characters.<br>**Note** The **service-policy output** *policy-map-name* command is not supported for applying a QoS policy map for distributed control plane services. |
| **Step 5** | **end**<br>**Example:**<br>`Router(config-cp)# end` | (Optional) Returns to privileged EXEC mode. |

# Verifying Aggregate Control Plane Services

To display information about the service policy attached to the control plane for aggregate CP services, complete the following steps.

### SUMMARY STEPS

1. **enable**
2. **show policy-map control-plane** [**all**] [**input** [**class** *class-name*] | output [class class-name]]
3. **exit**

### 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** | **show policy-map control-plane** [**all**] [**input** [**class** *class-name*] \| output [class class-name]]<br>**Example:**<br>`Router# show policy-map control-plane all` | Displays information about the control plane. Note the following points:<br>• **all** --(Optional) Service policy information about all QoS policies used in aggregate and distributed CP services.<br>• **input** --(Optional) Statistics for the attached input policy.<br>• **output** --(Optional) Statistics for the attached output policy.<br>• **class** *class-name* --(Optional) Name of the traffic class whose configuration and statistics are displayed. |
| **Step 3** | **exit**<br>**Example:**<br>`Router(config-cp)# exit` | (Optional) Exits privileged EXEC mode. |

#### Examples

The following example shows that the policy map TEST is associated with the control plane. This policy map polices traffic that matches the class map TEST, while allowing all other traffic (that matches the class map "class-default") to go through as is.

```
Router# show policy-map control-plane
```

```
Control Plane
Service-policy input:TEST
Class-map:TEST (match-all)
      20 packets, 11280 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:access-group 101
      police:
        8000 bps, 1500 limit, 1500 extended limit
        conformed 15 packets, 6210 bytes; action:transmit
        exceeded 5 packets, 5070 bytes; action:drop
        violated 0 packets, 0 bytes; action:drop
        conformed 0 bps, exceed 0 bps, violate 0 bps
Class-map:class-default (match-any)
      105325 packets, 11415151 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:any
```

# Verifying Distributed Control Plane Services

To display information about the service policy attached to the control plane to perform distributed CP services, complete the following steps.

### SUMMARY STEPS

1. **enable**
2. **show policy-map control-plane** [**all**][**slot** *slot-number*] [**input** [**class** *class-name*] | output [class class-name]]
3. **exit**

### 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** | **show policy-map control-plane** [**all**][**slot** *slot-number*] [**input** [**class** *class-name*] \| output [class class-name]]<br>**Example:**<br>`Router# show policy-map control-plane slot 2` | Displays information about the service policy used to apply distributed CP services on the router. Note the following points:<br>• **all** --(Optional) Service policy information about all QoS policies used in aggregate and distributed CP services.<br>• **slot** *slot-number* --(Optional) Service policy information about the QoS policy map used to perform distributed CP services on the specified line card.<br>• **input** --(Optional) Statistics for the attached input policy map.<br>• **output** --(Optional) Statistics for the attached output policy map.<br>• **class** *class-name* --(Optional) Name of the traffic class whose configuration and statistics are displayed. |

| | Command or Action | Purpose |
|---|---|---|
| Step 3 | **exit**<br>**Example:**<br>`Router# exit` | (Optional) Exits privileged EXEC mode. |

### Examples

The following example shows how to display information about the classes of CP traffic received from all interfaces on the line card in slot 1 to which the policy map TESTII is applied for distributed CP services. This policy map polices traffic that matches the traffic class TESTII, while allowing all other traffic (that matches the class map "class-default") to go through as is.

```
Router# show policy-map control-plane slot 1
Control Plane - slot 1
Service-policy input: TESTII (1048)
Class-map: TESTII (match-all) (1049/4)
        0 packets, 0 bytes
        5 minute offered rate 0 bps, drop rate 0 bps
        Match: protocol arp (1050)
        police:
            cir 8000 bps, bc 4470 bytes, be 4470 bytes
          conformed 0 packets, 0 bytes; actions:
            transmit
          exceeded 0 packets, 0 bytes; actions:
            drop
          violated 0 packets, 0 bytes; actions:
            drop
          conformed 0 bps, exceed 0 bps, violate 0 bps
Class-map: class-default (match-any) (1052/0)
        0 packets, 0 bytes
        5 minute offered rate 0 bps, drop rate 0 bps
        Match: any  (1053)
```

# Configuring Control Plane Policing to Mitigate Denial-of-Service Attacks

Apply control plane policing (CoPP) to RSVP packets to mitigate denial of service (DoS) attacks.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **permit** *protocol* {**any** | **host** {*address* | *name*}} {**any** | **host** {*address* | *name*}}
4. **access-list** *access-list-number* **permit** *protocol* {**tcd** | **udp**} {**any** | **host** {*source-addr* | *name*}} **eq** *port number* {**any** | **host** {*source-addr* | *name*}} **eq** *port number*
5. **class-map** *class-map-name*
6. **match access-group** *access-list-index*
7. **exit**
8. **policy-map** *policy-map-name*
9. **class** *class-map-name*
10. **police rate** *units* **pps**
11. **conform-action** *action*
12. **exit**
13. **exit**
14. **control plane** [**host** | **transit** | **cef-exception**]
15. **service-policy** {**input** | **output**} *policy-map-name*
16. **exit**
17. **exit**
18. **show control-plane** {**aggregate** | **cef-exception** | **counters** | **features** | **host** | **transit**}

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **access-list** *access-list-number* **permit** *protocol* {**any** \| **host** {*address* \| *name*}} {**any** \| **host** {*address* \| *name*}}<br>**Example:**<br>`Device(config)# access-list 140 permit 46 any any` | Configures an access list for filtering frames by protocol type. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | **access-list** *access-list-number* **permit** *protocol* {**tcd** \| **udp**} {**any** \| **host** {*source-addr* \| *name*}} **eq** *port number* {**any** \| **host** {*source-addr* \| *name*}} **eq** *port number*<br>**Example:**<br>`Device(config)# access-list 141 permit udp any eq 1699 any eq 1698` | Configures an access list for filtering frames by UDP protocol and matches only packets with a given port number. |
| **Step 5** | **class-map** *class-map-name*<br>**Example:**<br>`Device(config)# class-map match-any MyClassMap` | Creates a class-map and enters QoS class-map configuration mode. |
| **Step 6** | **match access-group** *access-list-index*<br>**Example:**<br>`Device(config-cmap)# match access-group 140` | Specifies access groups to apply to an identity policy. The range of valid values is 1-2799. |
| **Step 7** | **exit**<br>**Example:**<br>`Device(config-cmap)# exit` | Exits QoS class-map configuration mode and returns to global configuration mode. |
| **Step 8** | **policy-map** *policy-map-name*<br>**Example:**<br>`Device(config)# policy-map Policy1` | Specifies a service policy and enters QoS policy-map configuration mode. |
| **Step 9** | **class** *class-map-name*<br>**Example:**<br>`Device(config-pmap-)# class MyClassMap` | Enters QoS policy-map class configuration more |
| **Step 10** | **police rate** *units* **pps**<br>**Example:**<br>`Device(config-pmap-c)# police rate 10 pps` | Polices traffic destined for the control plane at a specified rate. |
| **Step 11** | **conform-action** *action*<br>**Example:**<br>`Device(config-pmap-c-police)# conform-action transmit` | (Optional) Specifies the action to take on packets that conform to the police rate limit and enters policy-map class police configuration mode. |
| **Step 12** | **exit**<br>**Example:**<br>`Device(config-pmap-c-police)# exit` | Exits policy-map class police configuration mode |

| | Command or Action | Purpose |
|---|---|---|
| **Step 13** | **exit**<br>**Example:**<br>`Device(config-pmap-)# exit` | Exits policy-map class configuration mode |
| **Step 14** | **control plane** [**host** \| **transit** \| **cef-exception**]<br>**Example:**<br>`Device(config)# control-plane` | Associates or modifies attributes (such as a service policy) that are associated with the control plane of the device and enters control plane configuration mode. |
| **Step 15** | **service-policy** {**input** \| **output**} *policy-map-name*<br>**Example:**<br>`Device(config-cp)# service-policy input Policy1` | Attaches a policy map to a control plane. |
| **Step 16** | **exit**<br>**Example:**<br>`Device(config-cp)# exit` | Exits control plane configuration mode and returns to global configuration mode. |
| **Step 17** | **exit**<br>**Example:**<br>`Device(config)# exit` | Exits global configuration mode returns to privileged EXEC mode. |
| **Step 18** | **show control-plane** {**aggregate** \| **cef-exception** \| **counters** \| **features** \| **host** \| **transit**}<br>**Example:**<br>`Device# show control-plane features` | Displays the configured control plane features |

# Configuration Examples for Control Plane Policing

## Example: Configuring Control Plane Policing on Input Telnet Traffic

The following example shows how to apply a QoS policy for aggregate control plane services to Telnet traffic that is received on the control plane. Trusted hosts with source addresses 10.1.1.1 and 10.1.1.2 forward Telnet packets to the control plane without constraint while allowing all remaining Telnet packets to be policed at the specified rate.

```
! Allow 10.1.1.1 trusted host traffic.
Device(config)# access-list 140 deny tcp host 10.1.1.1 any eq telnet

! Allow 10.1.1.2 trusted host traffic.
Device(config)# access-list 140 deny tcp host 10.1.1.2 any eq telnet
```

```
! Rate-limit all other Telnet traffic.
Device(config)# access-list 140 permit tcp any any eq telnet
! Define class-map "telnet-class."
Device(config)# class-map telnet-class

Device(config-cmap)# match access-group 140
Device(config-cmap)# exit
Device(config)# policy-map control-plane-in
Device(config-pmap)# class telnet-class
Device(config-pmap-c)# police 80000 conform transmit exceed drop
Device(config-pmap-c)# exit
Device(config-pmap)# exit
! Define aggregate control plane service for the active route processor.
Device(config)# control-plane
Device(config-cp)# service-policy input control-plane-in
Device(config-cp)# end
```

## Example: Configuring Control Plane Policing on Output ICMP Traffic

The following example shows how to apply a QoS policy for aggregate control plane services to Telnet traffic transmitted from the control plane. Trusted networks with source addresses 10.0.0.0 and 10.0.0.1 receive Internet Control Management Protocol (ICMP) port-unreachable responses without constraint while allowing all remaining ICMP port-unreachable responses to be dropped.

```
! Allow 10.0.0.0 trusted network traffic.
Device(config)# access-list 141 deny icmp 10.0.0.0 0.0.0.255 any port-unreachable

! Allow
10.0.0.1
 trusted network traffic.
Device(config)# access-list 141 deny icmp 10.0.0.1 0.0.0.255 any port-unreachable

! Rate-limit all other ICMP traffic.
Device(config)# access-list 141 permit icmp any any port-unreachable
Device(config)# class-map icmp-class

Device(config-cmap)# match access-group 141
Device(config-cmap)# exit
Device(config)# policy-map control-plane-out
! Drop all traffic that matches the class "icmp-class."
Device(config-pmap)# class icmp-class
Device(config-pmap-c)# drop
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# control-plane
! Define aggregate control plane service for the active route processor.
Device(config-cp)# service-policy output control-plane-out
Device(config-cp)# end
```

## Example: Configuring Control Plane Policing to Mitigate Denial-of-Service Attacks

The following example shows how to configure control plane policing (CoPP) to police RSVP packets at a specified rate and displays configured CoPP features.

```
Device> enable
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# access-list 140 permit 46 any any
Device(config)# access-list 141 permit adp any eq 1699 any eq 1698
Device(config)# class-map match-any MyClassMap
Device(config-cmap)# match access-group 140
Device(config-cmap)# match access-group 141
Device(config-cmap)# exit
Device(config)# policy-map Policy1
Device(config-pmap)# class MyClassMap
Device(config-pmap-c)# police rate 10 pps
Device(config-pmap-c-police)# conform-action transmit
Device(config-pmap-c-police)# exceed-action drop
Device(config-pmap-c-police)# exit
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# control-plane
Device(config-cp)# service-policy input Policy1
Device(config-cp)#
*Sep 14 08:07:39.898: %CP-5-FEATURE: Control-plane Policing feature enabled on Control plane
 aggregate path
Device(config-cp)#
Device(config-c p)# exit
Device(config)# exit
Device#
*Sep 14 08:09:04.154: %SYS-5-CONFIG_I: Configured from console by console
Device# show control-plane features
Total 1 features configured

Control plane aggregate path features :

--------------------------------------------------------
Control-plane Policing activated Sep 14 2012 08:0

--------------------------------------------------------
```

# Additional References

The following sections provide references related to the Control Plane Policing feature.

### Related Documents

| Related Topic | Document Title |
|---|---|
| QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | *Cisco IOS Quality of Service Solutions Command Reference* |
| QoS features overview | "Quality of Service Overview" module |
| MQC | "Applying QoS Features Using the MQC" module |
| Security features overview | "Control Plane Security Overview" module in the *Cisco IOS Security Configuration Guide: Securing the Control Plane* |
| Control plane policing in Cisco IOS Release 12.2(18)SXD1 and later releases | For Catalyst 6500 series switches, see the "Configuring Control Plane Policing (CoPP)" module.<br>For Cisco 7600 series routers, see the "Configuring Denial of Service Protection" module. |
| Enhanced RP protection | "ACL IP Options Selective Drop" module |

### Standards

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

### MIBs

| MIB | MIBs Link |
|---|---|
| • CISCO-CLASS-BASED-QOS-MIB<br>**Note** Supported only in Cisco IOS Release 12.3(7)T. | 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 |
|---|---|
| No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature. | -- |

### 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 Control Plane Policing

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 . An account on Cisco.com is not required.

*Table 5: Feature Information for Control Plane Policing*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Control Plane Policing | 12.2(18)S 12.3(4)T 12.3(7)T 12.0(29)S 12.2(18)SXD1 12.0(30)S 12.2(27)SBC 12.0(32)S 12.3(31)SB2 15.0(1)S | |

| Feature Name | Releases | Feature Information |
|---|---|---|
| | | The Control Plane Policing feature allows users to configure a quality of service (QoS) filter that manages the traffic flow of control plane packets to protect the control plane of Cisco IOS routers and switches against reconnaissance and denial-of-service (DoS) attacks.<br>For Release 12.2(18)S, this feature was introduced.<br>For Release 12.3(4)T, this feature was integrated into Cisco IOS Release 12.3(4)T, and the output rate-limiting (silent mode operation) feature was added.<br>For Release 12.3(7)T, the CISCO-CLASS-BASED-QOS-MIB was extended to manage control plane QoS policies, and the **police rate**command was introduced to support traffic policing on the basis of packets per second for control plane traffic.<br>For Release 12.0(29)S, this feature was integrated into Cisco IOS Release 12.0(29)S.<br>For Release 12.2(18)SXD1, this feature was integrated into Cisco IOS Release 12.2(18)SXD1.<br>For Release 12.0(30)S, this feature was modified to include support for distributed control plane services on the Cisco 12000 series Internet router.<br>For Release 12.2(27)SBC, this feature was integrated into Cisco IOS Release 12.2(27)SBC.<br>For Release 12.0(32)S, this feature was modified to include support for aggregate control plane services on the Cisco 10720 Internet router.<br>For Release 12.3(31)SB2, this feature was implemented on the Cisco 10000 series router for the PRE3. |

| Feature Name | Releases | Feature Information |
|---|---|---|
| | | For Release 15.0(1)S, this feature was integrated into Cisco IOS Release 15.0(1)S. |

CHAPTER 12

# Control Plane Protection

The Control Plane Protection feature is an extension of the policing functionality provided by the existing Control-Plane Policing feature. The Control-Plane Policing feature allows Quality of Service (QoS) policing of aggregate control-plane traffic destined to the route processor. The Control Plane Protection feature extends this policing functionality by allowing finer policing granularity.

The functionality added with Control Plane Protection includes a traffic classifier, which intercepts traffic and classifies it into three control-plane categories. New port-filtering and queue-thresholding features have also been added. The port-filtering feature provides for policing of packets going to closed or nonlistened TCP/UDP ports, while queue-thresholding limits the number of packets for a specified protocol that will be allowed in the control-plane IP input queue.

## 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 at the end of this module.

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

# Prerequisites for Control Plane Protection

- You understand the principles of Control-Plane Policing and how to classify control-plane traffic.
- You understand the concepts and general configuration procedure (class map and policy map) for applying QoS policies on a router.

For information about control plane policing and its capabilities, see the “Control Plane Policing” module.

For information about Cisco IOS QoS and the procedure for configuring QoS in your network using the Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC), see the *QoS: Modular QoS: Command-Line Interface Configuration Guide*.

# Restrictions for Control Plane Protection

### Control Plane Protection for IPv4

Control Plane Protection is restricted to IPv4 input path only.

### No Support for Direct ACL Configuration

The current release of Control Plane Protection does not support direct access control list (ACL) configuration in the control-plane subinterfaces, but rather can be configured using Modular QoS CLI (MQC) policies.

### Requires CEF

Control Plane Protection depends on Cisco Express Forwarding (CEF) for IP packet redirection. If you disable CEF globally, this will remove all active protect and policing policies configured on the control-plane subinterfaces. Aggregate control-plane interface policies will continue to function as normal.

### Control-plane Feature Policy Restriction

Policies applicable on the control-plane host subinterface are subject to the following restrictions:

- The port-filter feature policy supports only TCP/UDP-based protocols.
- The queue-thresholding feature policy supports only TCP/UDP-based protocols.

### No Support for Distributed or Hardware Switching Platforms

This release does not provide support for distributed or hardware switching platforms.

### Control-plane IP Traffic Classification Restrictions

The control-plane host subinterface only supports TCP/UDP-based host traffic. All IP packets entering the control-plane matching any of the following conditions are not classified any further and are redirected to the cef-exception subinterface:

- IP Packets with IP options.
- IP Packets with TTL less than or equal to 1.

### Protocols Auto-detected by the Port-filter

Some Cisco IOS TCP/UDP-based services, when configured, may not be auto-detected by the port-filter. That is, they do not get listed under the **show control-plane host open ports** output and they are not classified as an open port. This type of port must be manually added to the active port-filter class-map to be unblocked.

### Control-plane Policing Subinterface Restrictions

There are no restrictions on existing aggregate control-plane policing policies. New control-plane policing policies that are configured on host subinterface will not process ARP traffic since ARP traffic is processed at the cef-exception and aggregate interfaces.

# Information About Control Plane Protection

## Benefits of Control Plane Protection

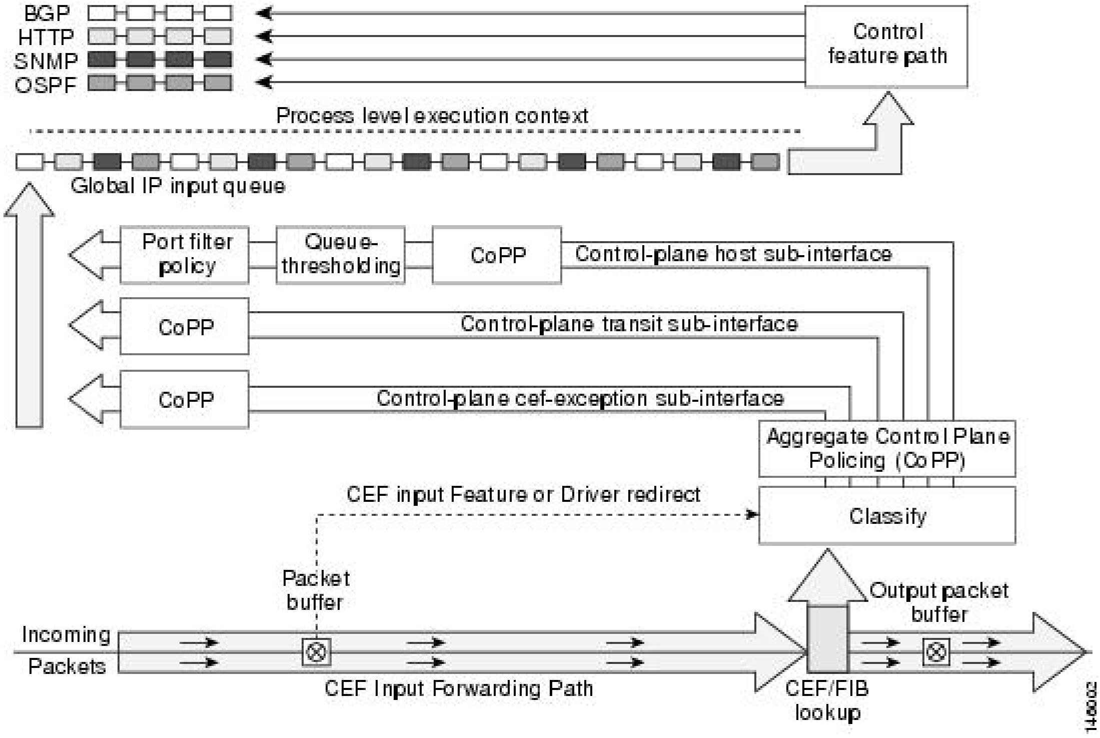
Configuring the Control Plane Protection feature on your Cisco router provides the following benefits:

- Extends protection against DoS attacks at infrastructure routers by providing mechanism for finer policing granularity for control-plane traffic that allows you to rate-limit each type individually.
- Provides a mechanism for early dropping of packets that are directed to closed or nonlistened IOS TCP/UDP ports.
- Provides ability to limit protocol queue usage such that no single protocol flood can overwhelm the input interface.
- Provides QoS control for packets that are destined to the control-plane of Cisco routers.
- Provides ease of configuration for control plane policies using MQC Infrastructure.
- Provides better platform reliability, security and availability.
- Provides dedicated control-plane subinterface for aggregate, host, transit and cef-exception control-plane traffic processing.
- Is highly flexible: permit, deny, rate-limit.
- Provides CPU protection so it can be used for important jobs, such as routing.

# Control Plane Protection Architecture

The figure below shows control-plane architecture with the Control Plane Protection feature.

*Figure 4: Control-plane Architecture with Control Plane Protection*


BGP
HTTP
SNMP
OSPF
Control feature path
Process level execution context
Global IP input queue
Port filter policy
Queue-thresholding
CoPP
Control-plane host sub-interface
CoPP
Control-plane transit sub-interface
CoPP
Control-plane cef-exception sub-interface
Aggregate Control Plane Policing (CoPP)
CEF input Feature or Driver redirect
Classify
Packet buffer
Output packet buffer
Incoming Packets
CEF Input Forwarding Path
CEF/FIB lookup
146002

The following sections describe the components of the Control Plane Protections feature.

# Control-plane Interface and Subinterfaces

Control Plane Policing (CoPP) introduced the concept of early rate-limiting protocol specific traffic destined to the processor by applying QoS policies to the aggregate control-plane interface. Control Plane Protection extends this control plane functionality by providing three additional control-plane subinterfaces under the top-level (aggregate) control-plane interface. Each subinterface receives and processes a specific type of control-plane traffic. The three subinterfaces are:

- **Control-plane host subinterface** . This interface receives all control-plane IP traffic that is directly destined for one of the router interfaces. Examples of control-plane host IP traffic include tunnel termination traffic, management traffic or routing protocols such as SSH, SNMP, BGP, OSPF, and EIGRP. All host traffic terminates on and is processed by the router. Most control plane protection features and policies operate strictly on the control-plane host subinterface. Since most critical router control plane services, such as routing protocols and management traffic, is received on the control-plane host subinterface, it is critical to protect this traffic through policing and protection policies. CoPP, port-filtering and per-protocol queue thresholding protection features can be applied on the control-plane host subinterface.

**Note** Non-IP based Layer 2 protocol packets such as ARP or CDP do not fall within the control-plane host subinterface. These packets are currently classified in the control-plane CEF-exception subinterface traffic.

- **Control-plane transit subinterface** . This subinterface receives all control-plane IP traffic that is software switched by the route processor. This means packets that are not directly destined to the router itself but rather traffic traversing through the router. Nonterminating tunnels handled by the router is an example of this type of control-plane traffic. Control Plane Protection allows specific aggregate policing of all traffic received at this subinterface.
- **Control-plane CEF-exception subinterface** . This control-plane subinterface receives all traffic that is either redirected as a result of a configured input feature in the CEF packet forwarding path for process switching or directly enqueued in the control plane input queue by the interface driver (i.e. ARP, L2 Keepalives and all non-IP host traffic). Control Plane Protection allows specific aggregate policing of this type of control plane traffic.

QoS policies attached on any of the control-plane interfaces or subinterfaces execute at interrupt level prior to packets being enqueued to the IP input queue and sent to the processor.

The transit and CEF-exception control plane subinterfaces exist in parallel to the control plane host subinterface. This release of Control Plane Protection allows for rate-limiting policies to be configured on these paths as Control Plane Policing extensions. The port-filtering and per-protocol queue thresholding features are not available on these control-plane subinterfaces.

All protection features in the control plane are implemented as MQC policies that operate using the control plane class-maps and policy-maps. New class-map and policy-map types have been created for the control plane port-filter and per-protocol queue-threshold features.

# Control-plane Port-filtering

The control-plane Port-filtering feature enhances control plane protection by providing for early dropping of packets directed toward closed or nonlistened IOS TCP/UDP ports on the router. The port-filter feature policy can be applied only to the control-plane host subinterface.

The port-filter maintains a global database of all open TCP and UDP ports on the router, including random ephemeral ports created by applications. The port database is dynamically populated with entries provided by the registered applications as they start listening on their advertised ports either by configuration of an application (that is SNMP) or initiation of an application (that is, TFTP transfer). An MQC class-map using the list of open ports can be configured and a simple drop policy can be applied to drop all packets destined to closed or nonlistened ports. Port-filter class-maps also support direct match of any user configured TCP/UDP port numbers.

# Control-plane Queue-thresholding

Control-plane protocol Queue-thresholding feature provides a mechanism for limiting the number of unprocessed packets a protocol can have at process-level. This feature can only be applied to the control-plane host subinterface. The intent of this feature is to prevent the input queue from being overwhelmed by any single protocol traffic. Per-protocol thresholding follows a protocol charge model. Each protocol's queue usage is limited such that no single mis-behaving protocol process can jam the interface hold queue. In this release, only a subset of TCP/UDP protocols can be configured for thresholding. Non-IP and Layer 2 protocols such as ARP and CDP cannot be configured. You can set queue limits for the following protocols:

- bgp—Border Gateway Protocol
- dns—Domain Name Server lookup
- ftp—File Transfer Protocol
- http—World Wide Web traffic
- igmp— Internet Group Management Protocol
- snmp—Simple Network Management Protocol
- ssh—Secure Shell Protocol
- syslog—Syslog Server
- telnet—Telnet
- tftp—Trivial File Transfer Protocol
- host-protocols—A wild card for all TCP/UDP protocol ports open on the router not specifically matched/configured

## Aggregate Control-plane Services

Control-plane Policing is an existing Cisco IOS feature that allows QoS policing of aggregate control-plane traffic destined to the route processor. The Control Plane Protection feature enhances protection for the router's control-plane by providing finer granularity of policing of traffic destined to the router's processor entering through any of the three control-plane subinterfaces. The CoPP feature is intended to be the first Control Plane Protection feature encountered by packets before any other features/policies. Existing (aggregate) Control-plane Policing policies will not be affected when the Control Plane Protection functionality is enabled. The aggregate Control-plane Policing policy will be applied on all control-plane traffic types. However, Control Plane Protection allows for additional and/or separate Control-plane Policing policies to be configured and applied on the different types of control-plane subinterfaces (host, transit, CEF-exception).

## Control Plane Protection Configuration

The CLI for control-plane (introduced with the Control Plane Policing feature) has been extended to allow for CoPP policies to be applied to individual control-plane subinterfaces (host, transit, CEF-exception). The command syntax for creating CoPP Service Policies remains the same. In addition, the MQC class-map and policy-map CLI was modified to allow for additional types. The port-filter and queue-threshold policy features available in the host subinterface uses these new class-map and policy-map "types".

CoPP leverages MQC to define traffic classification criteria and to specify configurable policy actions for the classified traffic. Traffic of interest must first be identified via class-maps, which are used to define packets for a particular traffic class. Once classified, enforceable policy actions for the identified traffic are created with policy-maps. The **control-plane** global command allows the control-plane service policies to be attached to the aggregate control-plane interface itself.

The CLI for configuring Control-plane Policing policies on the new control-plane subinterfaces remains basically the same as the CLI introduced for Control-plane Policing. The only difference is in how you apply or attach the CoPP policy to the different control-plane subinterfaces.

# How to Configure Control Plane Protection

## Defining Packet Classification Criteria for CoPP

Perform this task to define the packet classification criteria for CoPP.

### Before You Begin

Before you attach an existing QoS policy to the control-plane subinterface, you must first create the policy using the MQC to define a class map and policy map for control-plane traffic.

For information about how to classify traffic and create a QoS policy, see the *QoS: Modular QoS: Command-Line Interface Configuration Guide*.

**Note**

- The Control-plane Policing feature requires the MQC to configure packet classification and policing. Thus, restrictions that apply to MQC also apply to control-plane policing.
- Only the following classification (match) criteria are supported: standard and extended IP access lists (named or numbered) and the **match ip dscp** command, the **match ip precedence** command, and the **match protocol arp** command.
- The control-plane policing CLI does not support “type” extensions available with other protection features. This is to preserve backward-compatibility.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** [**match-any** | **match-all**] *class-map-name*
4. **match** {**access-group** | **name** *access-group-name*}

### 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** | **class-map** [**match-any** \| **match-all**] *class-map-name*<br>**Example:**<br>`Router(config)# class-map match-any control-plane-class` | Enables class map global configuration command mode used to create a traffic class.<br>• **match-any** —Specifies that one of the match criterion must be met for traffic entering the traffic class to be classified as part of the traffic class.<br>• **match-all** —Specifies that all match criterion must be met for traffic entering the traffic class to be classified as part of the traffic class.<br>• *class-map-name* —Specifies the user-defined name of the traffic class. Names can be a maximum of 40 alphanumeric characters. |
| **Step 4** | **match** {**access-group** \| **name** *access-group-name*}<br>**Example:**<br>`Router(config-cmap)# match access-group name cpp-igp-acl` | Specifies the match criteria for the class-map. |

# Defining a CoPP Service Policy

To define a service policy, use the policy-map global configuration command to specify the service policy name, and use the configuration commands to associate a traffic class that was configured with the class-map command, with the QoS action. The traffic class is associated with the service policy when the class command is used. You must issue the class command after entering policy-map configuration mode. After entering the class command, you are automatically in policy-map class configuration mode.

For information about how to classify traffic and create a QoS policy, see the *QoS: Modular QoS: Command-Line Interface Configuration Guide*.

Note

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control-plane interface.
- The Control-plane Policing feature requires the modular QoS command-line interface (CLI) (MQC) to configure packet classification and policing. Thus, restrictions that apply to MQC also apply to control-plane policing. Also, only two MQC actions are supported in policy maps - police and drop.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases, and only on the aggregate control-plane interface. Only input policing is available on the new control-plane host, transit and CEF-exception subinterfaces. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-name*
5. **police rate** [**burst-normal**] [**burst-max**] **conform-action action exceed-action action** [**violate-action** *action*]

## 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** *policy-map-name*<br>**Example:**<br>`Router(config)# policy-map control-plane-policy` | Enters policy map configuration mode to define a policy.<br>• *policy-map-name* —Name of a service policy map. The name can be a maximum of 40 alphanumeric characters. |
| **Step 4** | **class** *class-name*<br>**Example:**<br>`Router(config-pmap)# class control-plane-class` | Enters class map configuration mode, which is used to associate a service policy with a class.<br>• *class-name* —Name of a service policy class. The name can be a maximum of 40 alphanumeric characters. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | **police rate** [**burst-normal**] [**burst-max**] **conform-action action exceed-action action** [**violate-action** *action*]<br>**Example:**<br>`Router(config-pmap-c)# police rate 50000 pps conform-action transmit exceed-action drop` | To configure traffic policing, use the police command in policy-map class configuration mode or policy-map class police configuration mode.<br>• **rate —**Specifies the police rate. If the police rate is specified in pps, the valid value range is 1 to 2000000. If the police rate is specified in bps, the valid range of values is 8000 to 10000000000.<br>• **pps —**(Optional) Packets per second (pps) will be used to determine the rate at which traffic is policed.<br>• **conform-action action —**Action to take on packets that conform to the rate limit.<br>• **exceed-action action —**Action to take on packets that exceed the rate limit. |

# Entering Control Plane Configuration Mode

After you have created a class of traffic and defined the service policy for the control-plane, apply the policy to either the aggregate control-plane interface or one of the subinterfaces.

**Note**

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control-plane interface.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases, and only on the aggregate control-plane interface. Only input policing is available on the new control-plane host, transit and CEF-exception subinterfaces. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception**]

## 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** | **control-plane** [**host** \| **transit** \| **cef-exception**]<br>**Example:**<br>`Router(config)# control-plane` | Enters control-plane configuration mode to attach a QoS policy that manages CP traffic to specified control-plane subinterface:<br>• **host** —enters control-plane host subinterface configuration mode.<br>• **transit** —enters control-plane transit subinterface configuration mode.<br>• **cef-exception** —enters control-plane cef-exception subinterface configuration mode. |

# Applying CoPP Service Policy

Perform this task to apply CoPP service policies to a control-plane interface.

### Before You Begin

Before you attach an existing QoS policy to the control-plane, you must first create the policy by using MQC to define a class map and policy map for control-plane traffic.

For information about how to classify traffic and create a QoS policy, see the *QoS: Modular QoS: Command-Line Interface Configuration Guide*.

**Note**

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control-plane interface.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases, and only on the aggregate control-plane interface. Only input policing is available on the new control-plane host, transit and CEF-exception subinterfaces. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception**]
4. **service-policy** {**input** | **output**} *policy-map-name*

## 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** | **control-plane** [**host** \| **transit** \| **cef-exception**]<br>**Example:**<br>`Router(config)# control-plane host` | Attaches a QoS policy that manages CP traffic to a specified subinterface, and enters the control-plane configuration mode.<br>• **host** —applies policies to host control-plane traffic<br>• **transit** — applies policies to transit control-plane traffic<br>• **cef-exception** —applies policies to CEF-exception control-plane traffic |
| **Step 4** | **service-policy** {**input** \| **output**} *policy-map-name*<br>**Example:**<br>`Router(config-cp)# service-policy input control-plane-policy` | Attaches a QoS service policy to the control-plane.<br>• **input** —Applies the specified service policy to packets received on the control-plane.<br>• **output** —Applies the specified service policy to packets transmitted from the control-plane and enables the router to silently discard packets.<br>• *policy-map-name* —Name of a service policy map (created by using the **policy-map** command) to be attached. The name can be a maximum of 40 alphanumeric characters. |

# Configuring Port-filter Policy

You can apply the port-filter policy feature to the control-plane host subinterface to block traffic destined to closed or nonlistened TCP/UDP ports. New class-map and service-policy types have been created to accommodate the port-filter configuration. The classification and match criteria for the new port-filter class-maps supports only a constrained subset of the overall global MQC match criteria. Also, the actions supported by the new port-filter service policy is limited as well. that is only the drop action is supported

## Restrictions

- The classification and match criteria for the new port-filter class-maps support only a constrained subset of the overall global MQC match criteria.
- The actions supported by the new port-filter service policy is limited. Only the drop action is supported.
- The port-filter feature policy can only be attached on the control-plane host subinterface.
- Some IOS TCP/UDP-based services, when configured, may not be auto-detected by the port filter. That is, they do not get listed under the "show control plane host open ports" output and are not classified as an open port. This type of port must be manually added to the active port filter class-map to be unblocked when using the ‘closed-port’ match criteria.

There are three required steps to configure a port-filter policy:

## Defining Port-filter Packet Classification Criteria

Before you can attach a port-filter service policy to the control-plane host subinterface, you must first create the policy using the modified MQC to define a port-filter class-map and policy-map type for control-plane traffic.

A new MQC class-map type called *port-filter* was created for the port-filter feature. You must first create one or more port-filter class-map(s) before you can create your port-filter service policy. Your port-filter class-maps will separate your traffic into “classes” of traffic in which your service policy will define actions on.

**Note**

- The classification and match criteria for the new port-filter class-maps supports only a constrained subset of the overall global MQC match criteria. That is, only a subset of match protocol criteria is supported.
- Some IOS TCP/UDP-based services, when configured, may not be auto-detected by the port filter. That is, they do not get listed under the "show control plane host open ports" output and are not classified as an open port. This type of port must be manually added to the active port filter class-map to be unblocked when using the ‘closed-port’ match criteria.

### SUMMARY STEPS

1. **enable**
2. **class-map type port-filter** [**match-all** | **match-any**] **class name**
3. **match** { **closed-ports** | **not** | **port**} { **TCP** | **UDP**} 0-65535

### 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** | **class-map type port-filter** [**match-all** \| **match-any**] **class name**<br>**Example:**<br>`Router(config)# class-map type port-filter match-all pf-class` | Creates a class map used to match packets to a specified class and enables the port-filter class-map configuration mode.<br>• **match-all** —performs a logical AND on the match criteria<br>• **match-any** —performs a logical OR on the match criteria<br>• *class-name* —Name of a service policy class. The name can be a maximum of 40 alphanumeric characters. |
| **Step 3** | **match** { **closed-ports** \| **not** \| **port**} { **TCP** \| **UDP**} 0-65535<br>**Example:**<br>`Router(config-cmap)# match closed-ports` | Specifies the TCP/UDP match criteria for the class-map<br>• **closed-ports** —matches automatically on all closed-ports on the router<br>• **port** —allows you to manually specify a TCP/UDP port to match on.<br>• **TCP** —specifies a TCP port to match on<br>• **UDP** —specifies an UDP port to match on |

## Defining Port-filter Service Policy

You can define a port-filter service policy that provides additional control-plane protection. Defining this policy supports early dropping of packets that are directed toward closed on nonlistened TCP/UDP ports on the router.

To configure a Port-filter service policy, use the new policy-map type port-filter global configuration command to specify the port-filter service policy name, and use the following configuration commands to associate a port-filter traffic class that was configured with the class-map type port-filter command, with the port-filter drop action command. The port-filter traffic class is associated with the service policy when the class command is used. The class command must be issued after entering policy-map configuration mode. After entering the class command, you are automatically in policy-map class configuration mode.

**Note** The actions supported by the new port-filter service policy is limited. Only the drop action is supported.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map type port-filter** *policy-map-name*
4. **class** *class-name*
5. **drop**

### 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 port-filter** *policy-map-name*<br>**Example:**<br>`Router(config-pcmap)# policy-map type port-filter cppr-pf-policy` | Creates the port-filter service policy and enters the policy-map configuration mode.<br>• *policy-map-name* —Name of a service policy map. The name can be a maximum of 40 alphanumeric characters. |
| **Step 4** | **class** *class-name*<br>**Example:**<br>`Router(config-cmap)# class pf-class` | Associates a service policy with a class and enters class map configuration mode.<br>• *class-name* —Name of a service policy class. The name can be a maximum of 40 alphanumeric characters. |
| **Step 5** | **drop**<br>**Example:**<br>`Router (config-cmap)# drop` | Applies the port-filter service policy action on the class. |

## Applying Port-filter Service Policy to the Host Subinterface

Perform this task to apply port-filter service policies to a subinterface.

### Before You Begin

Before you attach a port-filter service policy to the control-plane host subinterface, you must first create the policy using MQC to define a class map and policy map for the required control-plane traffic.

**Note** The port-filter feature can only be applied on the control-plane host subinterface and only as input policy.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception**]
4. **service-policy type port-filter** {**input**} *port-filter-policy-map-name*

## 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 the global configuration mode. |
| **Step 3** | **control-plane** [**host** \| **transit** \| **cef-exception**]<br>**Example:**<br>`Router(config)# control-plane host` | Attaches a QoS policy that manages traffic to the control-plane host subinterface and enters the control-plane configuration mode.<br>**Note** Port-filter can only be applied to the host subinterface.<br>• **host** —enters the control-plane host subinterface configuration mode |
| **Step 4** | **service-policy type port-filter** {**input**} *port-filter-policy-map-name*<br>**Example:**<br>`Router(config-cp)# service-policy input cppr-pf-policy` | Attaches a QoS service policy to the control-plane host subinterface.<br>• **input** — Applies the specified service policy to packets received on the control-plane.<br>• **port-filter-policy-map-name** —Name of a port-filter service policy map (created using the policy-map type port-filter command) to be attached. The name can be a maximum of 40 alphanumeric characters. |

### Examples

The following example shows how to configure a port-filter policy to drop all traffic destined to closed or “nonlistened” TCP/UDP ports:

```
Router(config)# class-map type port-filter pf-class
Router(config-cmap)# match closed-ports
Router(config-cmap)# exit
Router(config)# policy-map type port-filter pf-policy
Router(config-pmap)# class pf-class
Router(config-pmap-c)# drop
Router(config-pmap-c)# end
Router#
```

The following example shows how to configure a port-filter policy to drop all traffic destined to closed or “nonlistened” ports except NTP.

```
Router(config)# class-map type port-filter pf-class
Router(config-cmap)# match not port udp 123
Router(config-cmap)# match closed-ports
Router(config-cmap)# exit
Router(config)# policy-map type port-filter pf-policy
Router(config-pmap)# class pf-class
Router(config-pmap-c)# drop
Router(config-pmap-c)# end
Router#
```

# Configuring Queue-threshold Policy

The Control Plane Protection feature includes a new queue-threshold policy feature that can be applied to the control-plane host subinterface. The queue-threshold feature allows you to limit the number of packets for a given higher level protocol allowed in the control-plane IP input queue. Much like the port-filter feature, new class-map and policy-map types have been created to accommodate the queue-threshold feature. As with the port-filter feature, the queue-threshold feature supports a very specific class-map and policy-map capabilities.

## Restrictions

- The classification and match criteria for the new queue-threshold class-maps supports only a constrained subset of the overall global MQC match criteria. That is, only a subset of match protocol option.
- The actions supported by the new queue-threshold service policy is limited. Only the queue-limit action is supported.
- The queue-threshold feature is supported only on the control-plane host subinterface as an input policy.

There are three steps required to configure a Queue-threshold policy:

## Defining Queue-threshold Packet Classification Criteria

You can define a queue-threshold service policy when you want to limit the number of unprocessed packets that a protocol can have at process level.

Before you can attach a queue-threshold service policy to the control-plane host subinterface, you must first create the policy using the modified MQC to define a queue-threshold class-map and policy-map type for control-plane traffic.

A new MQC class-map type called *queue-threshold* was created for the queue-threshold feature. You must first create one or more queue-threshold class-map(s) before you can create your queue-threshold service policy. Your queue-threshold class-maps will separate your traffic into "classes" of traffic in which your service policy will define actions on.

**Note** The classification and match criteria for the new queue-threshold class-map supports only a constrained subset of the overall global MQC match criteria. That is, only a subset of the match protocol criteria is supported.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map type queue-threshold [match-all | match-any]** *class name*
4. **match protocol [bgp|dns|ftp|http|igmp|snmp|ssh|syslog|telnet|tftp|host-protocols]**

## 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 the global configuration mode. |
| **Step 3** | **class-map type queue-threshold [match-all \| match-any]** *class name*<br>**Example:**<br>`Router(config)#class-map type queue-threshold match-all cppr-pf` | Applies a class map for the queue-threshold and enables the queue-threshold class-map configuration mode.<br>• **match-all** —performs a logical AND on the match criteria<br>• **match-any** —performs a logical OR on the match criteria<br>• *class-name* —Name of a service policy class. The name can be a maximum of 40 alphanumeric characters. |
| **Step 4** | **match protocol [bgp\|dns\|ftp\|http\|igmp\|snmp\|ssh\|syslog\|telnet\|tftp\|host-protocols]** | Specifies the upper layer protocol match criteria for the class-map. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config-cmap)# match protocol bgp` | • **bgp** Border Gateway Protocol<br>• **dns** —Domain Name Server lookup<br>• **ftp** —File Transfer Protocol<br>• **http** —World Wide Web traffic<br>• **igmp** —Internet Group Management Protocol<br>• **snmp** —Simple Network Management Protocol<br>• **ssh** —Secure Shell Protocol<br>• **syslog** —Syslog Server<br>• **telnet** —Telnet<br>• **tftp** —Trivial File Transfer Protocol<br>• **host-protocols** —any open TCP/UDP port on the router. |

## Defining a Queue-threshold Service Policy

To configure a queue-threshold service policy, use the new policy-map type called queue-threshold global configuration command to specify the queue-threshold service policy name, and use the following configuration commands to associate a queue-threshold traffic class that was configured with the class-map type queue-threshold command, with the queue-threshold queue-limit action command. The queue-threshold traffic class is associated with the service policy when the class command is used. The class command must be issued after entering policy-map configuration mode. After entering the class command, you are automatically in policy-map class configuration mode.

**Note** The actions supported by the new queue-threshold service policy is limited. Only the queue-limit action is supported.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map type queue-threshold** *policy-name*
4. **class** *class-name*
5. **queue-limit** *number*

### 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 the global configuration mode. |
| **Step 3** | **policy-map type queue-threshold** *policy-name*<br>**Example:**<br>`Router(config)# policy-map type queue-threshold cppr-qt-policy` | Enables the queue-threshold service policy configuration mode.<br>- *policy-name* —Name of a service policy map. The name can be a maximum of 40 alphanumeric characters. |
| **Step 4** | **class** *class-name*<br>**Example:**<br>`Router(config-pcmap)# class qt-class` | Enters class map configuration mode used to associate a service policy with a class.<br>- *class-name* —Name of a service policy class. The name can be a maximum of 40 alphanumeric characters. |
| **Step 5** | **queue-limit** *number*<br>**Example:**<br>`Router(config-cmap) #queue-limit 75` | Applies the queue-threshold service policy action on the class.<br>**Note** Queue limit range is 0 to 255. |

## Applying a Queue-threshold Policy to the Host Subinterface

Perform this task to apply queue-threshold service policies to the control-plane host subinterface.

### Before You Begin

Before you attach a queue-threshold service policy to the control-plane host subinterface, you must first create the policy by using MQC to define a class map and policy map for the required control-plane traffic.

**Note** The queue-threshold feature can only be applied on the control-plane host subinterface as an input policy.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception**]
4. **service-policy type queue-threshold** {**input**} *queue-threshold-policy-map-name*

### 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 the global configuration mode. |
| Step 3 | **control-plane** [**host** \| **transit** \| **cef-exception**]<br>**Example:**<br>`Router(config)# control-plane host` | Attaches a QoS queue-threshold policy that manages traffic to the host subinterface and enters control-plane configuration mode.<br>• **host** —Enters the control-plane host subinterface configuration mode.<br>**Note** queue-threshold can only be applied to the host subinterface. |
| Step 4 | **service-policy type queue-threshold** {**input**} *queue-threshold-policy-map-name*<br>**Example:**<br>`Router(config-cp)# service-policy input cppr-qt-policy` | Attaches a QoS service policy to the control-plane.<br>• **input** —Applies the specified service policy to packets received on the control-plane.<br>• *queue-threshold-policy-map-name* —Name of a queue-threshold service policy map (created using the policy-map type queue-threshold command) to be attached. The name can be a maximum of 40 alphanumeric characters. |

### Examples

The following example shows how to configure a queue-threshold policy to set the queue limit for SNMP protocol traffic to 50, telnet traffic to 50, and all other protocols to 150.

```
Router(config)# class-map type queue-threshold qt-snmp-class
Router(config-cmap)# match protocol snmp
Router(config-cmap)# class-map type queue-threshold qt-telnet-class
Router(config-cmap)# match protocol telnet
Router(config-cmap)# class-map type queue-threshold qt-other-class
Router(config-cmap)# match host-protocols
Router(config-cmap)# exit
Router(config)# policy-map type queue-threshold qt-policy
Router(config-pmap)# class qt-snmp-class
Router(config-pmap-c)# queue-limit 50
Router(config-pmap-c)# class qt-telnet-class
Router(config-pmap-c)# queue-limit 50
Router(config-pmap-c)# class qt-other-class
Router(config-pmap-c)# queue-limit 150
Router(config-pmap-c)# end
Router#
```

# Verifying Control Plane Protection

Use the **show policy-map control-plane** command to verify Control Plane Protection configurations and to view statistics for control-plane service policies.

To display information about the service policy attached to the control-plane, perform the following optional steps.

## SUMMARY STEPS

1. **enable**
2. **show policy-map** [**type** *policy-type*] **control-plane** [**pfx** | **slot** *slot number*] [**all**] [**host** | **transit** | **cef-exception**] [{**input** | **output**} [**class** *class-name*]]

## 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** | **show policy-map** [**type** *policy-type*] **control-plane** [**pfx** \| **slot** *slot number*] [**all**] [**host** \| **transit** \| **cef-exception**] [{**input** \| **output**} [**class** *class-name*]]<br>**Example:**<br>`Router# show policy-map control-plane all` | Displays information about the control-plane.<br>• **policy-type —** Specifies policy-map type that you want statistics for (i.e. port-filter or queue-threshold)<br>• **pfx —** Does not apply to Control Plane Protection feature.<br>• **slot —** Does not apply to Control Plane Protection feature<br>• **all —**Information for all control plane interfaces.<br>• **host —**Policy-map and class-map statistics for the host path.<br>• **transit —**Policy-map and class-map statistics for transit path.<br>• **cef-exception —**Policy-map and class-map statistics for CEF-exception path.<br>• **input —**Statistics for the attached input policy will be displayed. |

| | Command or Action | Purpose |
|---|---|---|
| | | • **output** —Statistics for the attached output policy will be displayed.<br>• **class** *class name* —Name of class whose configuration and statistics are to be displayed. |

## Examples

The following example shows that the aggregate CoPP policy map named “copp-transit-policy” is associated with the control-plane transit subinterface and displays the statistics for that policy:

```
Router# show policy-map control-plane transit

 control-plane Transit
  Service-policy input: copp-transit-policy
    Class-map: copp-transit-class (match-all)
      8 packets, 592 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
      police:
          rate 2000 pps, burst 488 packets
        conformed 8 packets; actions:
          transmit
        exceeded 0 packets; actions:
          drop
        conformed 0 pps, exceed 0 pps
    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
```

The following example shows that the policy map “TEST” is associated with the aggregate control-plane interface. This policy map polices traffic that matches the class map “TEST,” while allowing all other traffic (that matches the class map “class-default”) to go through as is.

```
Router# show policy-map control-plane

control-plane
Service-policy input:TEST
Class-map:TEST (match-all)
      20 packets, 11280 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:access-group 101
      police:
        8000 bps, 1500 limit, 1500 extended limit
        conformed 15 packets, 6210 bytes; action:transmit
        exceeded 5 packets, 5070 bytes; action:drop
        violated 0 packets, 0 bytes; action:drop
        conformed 0 bps, exceed 0 bps, violate 0 bps
Class-map:class-default (match-any)
      105325 packets, 11415151 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:any
```

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | *Cisco IOS Quality of Service Solutions Command Reference* |
| QoS feature overview | “Quality of Service Overview” module |

### Standards

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

### MIBs

| MIB | MIBs Link |
|---|---|
| • CISCO-CLASS-BASED-QOS-MIB<br>**Note** Supported only in Cisco IOS Release 12.3(7)T. | 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 |
|---|---|
| None | — |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content. | http://www.cisco.com/techsupport |

# Feature Information for Control Plane Protection

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 . An account on Cisco.com is not required.

*Table 6: Feature Information for Control Plane Protection*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Control Plane Protection | 12.4(4)T | The Control Plane Protection feature is an extension of the policing functionality provided by the existing Control-Plane Policing feature. The Control-Plane Policing feature allows QoS policing of aggregate control-plane traffic destined to the route processor. The Control Plane Protection feature extends this policing functionality by allowing finer policing granularity.<br>The following commands were introduced or modified: **class-map**, **control-plane**, **show policy-map control-plane** |

CHAPTER 13

# Control Plane Logging

The Cisco IOS Control Plane Protection features allow you to filter and rate-limit the packets that are going to the router's control plane, and discard malicious and or error packets. The addition of the Control Plane Logging feature enables logging of the packets that are dropped or permitted by these features. You can turn on logging for all or some packets that are processed by the control plane, without feature or class restrictions, or you can enable logging for specific Control Plane Protection features such as control plane policing, port-filtering, and queue-thresholding. The Control Plane Logging feature provides the logging mechanism that is needed to efficiently deploy, monitor, and troubleshoot Control Plane Protection features.

## 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 at the end of this module.

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

## Prerequisites for Control Plane Logging

- You understand the principles of control plane policing and how to classify control-plane traffic.

- You understand the concepts and general configuration procedures for control plane protection, including control plane policing, port-filtering, and queue-threshold.
- You understand the concepts and general configuration procedure for applying QoS policies on a router (class map and policy map).

For information about control plane policing and its capabilities, see the “Control Plane Policing” module.

For information about control plane protection and its capabilities, see the “Control Plane Protection” module.

For information about Cisco IOS QoS and the procedure for configuring QoS in your network using the Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC), see the *QoS: Modular QoS: Command-Line Interface Configuration Guide*.

# Restrictions for Control Plane Logging

- The Control Plane Logging feature logs control-plane packets only. This feature does not log data-plane traffic that traverses the router on non-control-plane interfaces.
- The Control Plane Logging feature logs IPv4 packets only. IPv6 packet logging is not supported.
- Control plane logging is supported only on platforms that support control plane protection.
- Packets permitted or dropped by the Management Plane Protection (MPP) feature can be logged only via the Global Control Plane Logging mechanism. Feature-specific or class-specific control plane logging cannot be used to log MPP traffic.
- Global control plane logging can log only dropped or error packets on the aggregate control-plane interface as a result of a control plane policing policy applied to the aggregate interface. To log allowed packets, you must apply the global control-plane logging policy to the host, transit, or cef-exception control-plane subinterface, or you must use feature-specific or class-specific logging.
- A packet that passes through the control plane can be logged only once using this feature. The state printed in the log message (PERMIT or DROP) is the final state of the packet on the control plane. For example, if there is a control-plane protection policy on the aggregate control-plane interface and another on the host control-plane subinterface, with logging enabled on both, a packet that is allowed by both features will be logged only once (with a state of PERMIT). So a state of PERMIT when logged for a packet means that the packet was allowed by all control-plane protection features.
- Although logging control-plane traffic provides valuable insight into the details of control-plane traffic, logging excessive control-plane traffic might result in an overwhelming number of log entries and possibly high router CPU usage. Use control plane logging for short periods of time and only when needed to help classify, monitor, and troubleshoot control-plane traffic and features.

# Information About Control Plane Logging

To configure the Control Plane Logging feature, you should understand the following concepts:

## Global Control Plane Logging

Global Control Plane Logging is a feature that allows logging of all or some packets processed by the control plane, without feature or class restrictions. This can be used to log all, or a subset of, traffic permitted or dropped by the Control Plane Protection Features. Packets to be logged can be filtered based on the basis of multiple match criteria (for example, input interface, source IP address, or destination IP address).

Logging policies can also log packets on the basis of the action taken on them (that is, dropped or permitted) by control plane features (that is, control plane policing, port-filtering or per-protocol queue-thresholding). Packets that are dropped by the control-plane infrastructure because of checksum errors can also be filtered and logged. If you have not specified the kind of packet to be logged via the "permitted," "dropped," or "error" action match criteria, all packets (permitted, dropped, and error) will be considered for logging.

By default, the log messages contain source IP address, destination IP address, protocol name (IP/TCP/UDP), action (permit, drop, error), and port number. Additionally, there are options that can be configured with the log action that can enable logging of other fields in the IP header as well, such as TTL and packet length. There is also an option to configure the rate-limit interval for which log messages are created; that is, the interval between the logging of two messages.

The Global Control Plane Logging feature is configured using new MQC class-map, policy-map, and service-policy types and can be applied on the aggregate control-plane interface or on a specific control-plane subinterface (that is, host, transit, or cef-exception).

## Feature-Specific or Class-Specific Logging

Feature-specific or class-specific logging tracks only packets that match a specific class and that are acted upon by a specific control plane protection feature (that is, control plane policing, port-filtering, or per-protocol queue-thresholding). This type of logging differs from global logging, which allows you to log all packets on a control-plane interface. With global logging, traffic that matches individual classes within a control plane protection feature policy cannot be distinguished. Global logging, for example, can log only all packets dropped on a control-plane interface as a whole. However, with feature-specific or class-specific logging, packets that match a specific class and that are acted upon by a specific control plane protection feature will be separated out. Feature-specific or class-specific logging may be most valuable during the initial stages of control plane protection deployment, when there is a need to know details about packets that match a specific class. For example, knowing what traffic is hitting your class-default class would help in modifying your class maps or policy maps to account for stray packets or for determining characteristics of an attack.

Feature-specific or class-specific logging provides feature-specific logging, making it possible to log packets for a specific feature on a specific control-plane interface (for example, port-filtering on the control-plane host interface).

Feature-specific or class-specific logging allows logging of packets that pass through a class map in a control plane protection feature service policy applied to a control-plane interface. When a feature, such as control plane policing, is applied on a control-plane interface, feature-specific or class-specific logging can be added as one of the actions to be performed on a class defined in the feature policy map. When logging is added as an action for a class inside a policy map, all packets that match that class will be logged. The only packets filtered are those that the feature class map supports. There is no further classification done for logging specifically. The **log** action keyword can be added by itself without any other policing actions defined in the class, or it can be added in addition to the police or drop action defined in the class. When the **log** keyword is added as an action for a class inside a policy map, all packets (permitted and/or dropped) that match the class will be logged.

By default, the log messages contain source IP address, destination IP address, protocol name (IP/TCP/UDP), action (permit, drop, error), and port number. Additionally, there are options that can be configured with the log action that can enable logging of other fields in the IP header as well, such as TTL and packet length. There is also an option to configure the rate-limit interval for which log messages are created; that is, the interval between the logging of two messages.

## Global Logging Configuration

To support global control plane logging, new MQC class-map, policy-map, and service-policy types were created. Policy-map type logging is used only for global control plane logging policies. Class-map type logging is used to classify what type of control-plane traffic you want to log. The logging type class maps support a subset of generic QoS match criteria and some control-plane-specific match criteria. The supported match criteria are as follows:

- input-interface
- IPv4 source IP address
- IPv4 destination IP address
- packets dropped
- packets permitted
- packets error

If one of the packet-action filters, packets dropped, packets permitted, or packets error, is not specified, all matching packets will be logged irrespective of the action taken on them (permitted or dropped).

Also, in a logging type policy map, the only action supported is log. The configuration and behavior of the **log** action keyword are the same in global logging and feature-specific or class-specific logging. The available options for the **log** action keyword are as follows:

- interval—Sets packet logging interval.
- ttl—Logs ttl for IPv4 packets.
- total-length—Logs packet length for IPv4 packets.

**Note** Logging policies can be applied to the control plane, control-plane host, control-plane transit, and control-plane cef-exception interfaces.

# How to Configure Logging on a Control Plane Interface

## Defining Packet Logging Classification Criteria for Global Logging

When configuring global logging, you must first define the packet logging classification criteria.

**Note** You can apply global logging policies on control plane interfaces only.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** [**type** {**stack** | **access-control** | **port-filter** | **queue-threshold** | **logging**}] [**match-all** | **match-any**] *class-map-name*
4. **match** [**input-interface** |**ipv4source-address**| **ipv4destination-address** | **notinput-interface** | **packets permitted** | **packets dropped** | **packets error**]
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** | **class-map** [**type** {**stack** \| **access-control** \| **port-filter** \| **queue-threshold** \| **logging**}] [**match-all** \| **match-any**] *class-map-name*<br>**Example:**<br>`Router(config)# class-map type logging match-all log-class` | Creates a class map used to match packets to a specified class and enters class-map configuration mode. The following keywords and arguments can be used for control plane logging:<br>• **type** — (Optional) Identifies the class-map type. Use the **logging keyword**for control plane logging configurations.<br>• **match-all** — (Optional) Performs a logical AND on the match criteria.<br>• **match-any** — (Optional) Performs a logical OR on the match criteria.<br>• *class-map name* — Name of a class. The name can be a maximum of 40 alphanumeric characters. |
| **Step 4** | **match** [**input-interface** \|**ipv4source-address**\| **ipv4destination-address** \| **notinput-interface** \| **packets permitted** \| **packets dropped** \| **packets error**] | Defines the match criteria for the logging class map. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config-cmap)# match packets dropped` | |
| **Step 5** | **end**<br>**Example:**<br>`Router(config-cmap)# end` | Exits class-map configuration mode and returns to privileged EXEC mode. |

# Defining the Logging Policy Map for Global Logging

After you define packet logging criteria for global logging, you must define the logging policy map.

To configure global logging policy maps, use the new **policy-map type logging** configuration command. Then, use the **class** command, to associate a logging class-map that was configured with the **class-map type logging** command, with the logging policy map. Use the **log** keyword to configure the log action for the class that you associated with the policy map. The **class** command must be issued after entering the policy-map configuration mode. After entering the **class** command, you are automatically in policy-map class configuration mode. The action **log** can be configured while in policy-map class configuration mode.

**Note** You can apply global logging policies on control plane interfaces only.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** [**type** {**stack** | **access-control** | **port-filter** | **queue-threshold** | **logging**}] *policy-map-name*
4. **class** *class-name*
5. **log** [**interval** *seconds* **total-length ttl**]
6. **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** | **policy-map** [**type** {**stack** \| **access-control** \| **port-filter** \| **queue-threshold** \| **logging**}] *policy-map-name*<br>**Example:**<br>`Router(config)# policy-map type logging log-policy` | Creates the logging service policy and enters policy-map configuration mode.<br>• **type** — (Optional) Identifies the policy-map type. Use the **logging** keyword for control plane logging configurations.<br>• *policy-map-name* — Name of a policy map. The name can be a maximum of 40 alphanumeric characters. |
| **Step 4** | **class** *class-name*<br>**Example:**<br>`Router(config-pmap)# class log-class` | Associates a class with a policy map and enters class-map configuration mode.<br>• *class-name* — Name of a class of type logging. The name can be a maximum of 40 alphanumeric characters. |
| **Step 5** | **log** [**interval** *seconds* **total-length ttl**]<br>**Example:**<br>`Router(config-pmap-c)# log interval 1000` | Applies the log action to the logging class. With this command, you can enter the following optional parameters:<br>• **interval** *seconds* — (Optional) Sets packet logging interval.<br>• **total-length** — (Optional) Logs packet length for IPv4 packets.<br>• **ttl** — (Optional) Logs ttl for IPv4 packets. |
| **Step 6** | **end**<br>**Example:**<br>`Router(config-pmap-c)# end` | Exits from class-map configuration mode and returns to privileged EXEC mode. |

# Creating a Logging Service Policy on a Control Plane Interface for Global Logging

After you define the logging service policy, you must apply the policy to a specific control plane interface.

**Note** You can apply global logging policies on control plane interfaces only.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception** | **cr**]
4. **service-policy type logging input logging-policy-map-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. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **control-plane** [**host** \| **transit** \| **cef-exception** \| **cr**]<br>**Example:**<br>`Router(config)# control-plane host` | Enters control-plane configuration mode.<br>• **host** — (Optional) Applies policies to control-plane host subinterface.<br>• **transit** — (Optional) Applies policies to control-plane transit subinterface.<br>• **cef-exception** — (Optional) Applies policies to control-plane cef-exception subinterface.<br>• **cr** — (Optional) Applies policies to all control-plane interfaces. |
| **Step 4** | **service-policy type logging input logging-policy-map-name**<br>**Example:**<br>`Router(config-cp)# service-policy type logging input log-policy` | Applies a logging policy to a control-plane interface.<br>• **input** — Applies the specified service policy to packets received on the control plane.<br>• *logging-policy-map-name* — Name of a logging policy map (created by using 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-cp)# end` | Exits control-plane configuration mode and returns to privileged EXEC mode. |

# Configuring Feature-Specific or Class-Specific Logging

Feature-specific or class-specific control plane logging is implemented as an integrated part of Cisco's Control Plane Protection features, such as per-protocol queue-thresholding, port-filter, or control plane policing, as an action within their respective policy maps. To enable feature-specific or class-specific control plane logging, the log action should be added to the existing Control Plane Protection feature policy map.

The default behavior for a policy with the log action is to log matching packets. By default, the log messages contain source IP address, destination IP address, protocol name (IP/TCP/UDP), action (permit, drop, error), and port number. Additionally, there are options that can be configured with the log action that can enable logging of other fields in the IP header as well, such as TTL and packet length. There is also an option to configure the rate-limit interval for which log messages are created, that is the interval between the logging of two messages.

The additional options for the **log** action keyword are as follows:

- interval—Sets packet logging interval.
- ttl—Logs ttl for Ipv4 packets.
- total-length—Logs packet length for IPv4 packets.

**Note** The log action can be added only to policy maps of control-plane protection features, which are control plane policing, port-filtering, and queue-thresholding.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** [**type** | { **stack** | **access-control** | **port-filter** | **queue-threshold** | **logging**}] *policy-map-name*
4. class *class-name*
5. **log [ interval** *seconds* **| total-length | ttl ]**
6. **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 | **policy-map** [**type** \| { **stack** \| **access-control** \| **port-filter** \| **queue-threshold** \| **logging**}] *policy-map-name*<br>**Example:**<br>`Router(config)# policy-map type queue-threshold qt-policy` | Creates a policy map and enters policy-map configuration mode.<br>• **type** — (Optional) Specifies the service policy type.<br>• **port-filter** — (Optional) Enters the policy map for the port-filter feature.<br>• **queue-threshold** — (Optional) Enters the policy map for the queue-threshold feature.<br>• **logging** — (Optional) Enters policy-map configuration mode for the control plane logging feature.<br>• *policy-map-name* — Name of the policy map. The name can be a maximum of 40 alphanumeric characters. |
| Step 4 | class *class-name*<br>**Example:**<br>`Router(config-pmap)# class qt-host` | Associates a class with a policy and enters class map configuration mode. |
| Step 5 | **log** [ **interval** *seconds* \| **total-length** \| **ttl** ]<br>**Example:**<br>`Router(config-pmap-c)# log interval 1000` | Applies the log action to the service-policy class. You can configure the following additional parameters:<br>• **interval** *seconds* —(Optional) Sets packet logging interval.<br>• **total-length** —(Optional) Logs packet length for IPv4 packets.<br>• **ttl** —(Optional) Logs ttl for IPv4 packets. |
| Step 6 | **end**<br>**Example:**<br>`Router(config-pmap-c)# end` | Exits class-map configuration mode and returns to privileged EXEC mode. |

# Verifying Control Plane Logging Information

You can verify control plane logging for both global logging configurations and feature-specific or class-specific configurations.

To display active control plane logging information for global logging, perform the following optional steps.

### SUMMARY STEPS

1. **enable**
2. **show policy-map type logging control-plane [host | transit | cef-exception | cr]**
3. show policy-map [type *policy-type*] control-plane [host | transit | cef-exception | all | cr]

### 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 | **show policy-map type logging control-plane [host \| transit \| cef-exception \| cr]**<br>**Example:**<br>`Router# show policy-map type` | Display information for global control plane logging. |
| Step 3 | show policy-map [type *policy-type*] control-plane [host \| transit \| cef-exception \| all \| cr]<br>**Example:**<br>`Router# show policy-map type logging control-plane host` | Display information for feature-specific or class-specific control plane logging.<br>**Note** The example shows feature-specific or class-specific logging enabled on a port-filter policy. |

# Verification Examples for Control Plane Logging

## Sample Output for a Global Logging Configuration

The following output displays the global logging service policy that was just added to the control-plane host feature path interface:

```
Router# show policy-map type logging control-plane host

 Control Plane Host
  Service-policy logging input: cpplog-host-policy
    Class-map: cpplog-host-map (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:  packets dropped
        0 packets, 0 bytes
        5 minute rate 0 bps
      Match:  packets permitted
        0 packets, 0 bytes
        5 minute rate 0 bps
    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
```

## Sample Output for a Feature-Specific or Class-Specific Configuration

The following output displays the logging policy map that was just added to the control-plane host feature path interface:

```
Router# show policy-map cpp-policy

  Policy Map cpp-policy
    Class cppclass-igp
    Class cppclass-management
     police rate 250 pps burst 61 packets
       conform-action transmit
       exceed-action drop
    Class cppclass-monitoring
     police rate 100 pps burst 24 packets
  conform-action transmit
       exceed-action drop
Class cppclass-undesirable
      drop
      log interval 5000
    Class class-default
     police rate 50 pps burst 12 packets
       conform-action transmit
       exceed-action drop
```

## Sample Log Output

The following example shows log output for a configuration that sends IP traffic to the router:

```
Router#
%CP-6-IP: PERMIT ttl=59 length=20 209.165.200.225 -> 209.165.200.254
%CP-6-IP: PERMIT ttl=59 length=20 209.165.200.225 -> 209.165.200.254
%CP-6-IP: PERMIT ttl=59 length=20 209.165.200.225 -> 209.165.200.254
%CP-6-IP: PERMIT ttl=59 length=20 209.165.200.225 -> 209.165.200.254
```

The following is a description of the log information displayed in the preceeding example:

- IP denotes the kind of traffic received.
- PERMIT means that no control-plane feature dropped the packet.
- ttl gives the ttl value in the IP header.
- length gives the total-length field in the IP header.
- 209.165.200.225 is the source IP address.
- 209.165.200.254 is the destination IP address.

The following example shows log output for a configuration that sends TCP traffic to the router:

Router#

```
%CP-6-TCP: PERMIT ttl=59 length=40 209.165.200.225(18611) -> 209.165.200.254(23)
%CP-6-TCP: PERMIT ttl=59 length=40 209.165.200.225(18611) -> 209.165.200.254(23)
%CP-6-TCP: PERMIT ttl=59 length=40 209.165.200.225(18611) -> 209.165.200.254(23)
%CP-6-TCP: PERMIT ttl=59 length=40 209.165.200.225(18611) -> 209.165.200.254(23)
```

The following is a description of the log information displayed in the preceding example:

- TCP denotes the kind of traffic received.
- PERMIT means that no control-plane feature dropped the packet.
- ttl gives the ttl value in the IP header.
- length gives the total-length field in the IP header.
- 209.165.200.225 is the source IP address.
- 18611 is the source TCP port.
- 209.165.200.254 is the destination IP address.
- 23 is the destination TCP port.

# Configuration Examples for Control Plane Logging

This section provides the following configuration examples:

## Configuring Global Control Plane Logging for Dropped and Permitted Packets Example

The following example shows how to configure a global control-plane logging service policy to log all dropped and permitted packets that hit the control-plane host feature path only, regardless of the interface from which the packets enter the router. Also, the router rate-limits the log messages to one every 5 seconds.

```
! Define a class map of type logging to specify what packets will be logged.
Router# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# class-map type logging match-any cpplog-host-map
Router(config-cmap)# match packets dropped
Router(config-cmap)# match packets permitted
Router(config-cmap)# exit

! Define a policy map of type logging using your logging class map and rate-limit log
  messages to one every 5 seconds.
Router(config)# policy-map type logging cpplog-host-policy
Router(config-pmap)# class cpplog-host-map
Router(config-pmap-c)# log interval 5000
Router(config-pmap-c)# exit
Router(config-pmap)# exit

! Apply the new logging policy map to the control-plane host feature path interface.
Router(config)# control-plane host
Router(config-cp)# service-policy type logging input cpplog-host-policy
Router(config-cp)# end
Router#
Aug  8 17:57:57.359: %SYS-5-CONFIG_I: Configured from console by cisco on console
Router#
```

The following output displays the logging policy map that was just added to the control-plane host feature path interface:

```
Router# show policy-map type logging control-plane host

 Control Plane Host
  Service-policy logging input: cpplog-host-policy
    Class-map: cpplog-host-map (match-any)
```

```
        0 packets, 0 bytes
        5 minute offered rate 0 bps, drop rate 0 bps
        Match:  packets dropped
          0 packets, 0 bytes
          5 minute rate 0 bps
        Match:  packets permitted
          0 packets, 0 bytes
          5 minute rate 0 bps
        log interval 5000
      Class-map: class-default (match-any)
        0 packets, 0 bytes
        5 minute offered rate 0 bps, drop rate 0 bps
        Match: any
```

## Configuring Global Control Plane Logging for Dropped Packets Example

The following example shows how to configure a global control-plane logging service policy to log all dropped packets that come from GigabitEthernet interface 0/3 that hit the aggregate control-plane interface.

```
! Define a class map of type logging to specify what packets will be logged.
Router# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# class-map type logging match-all cpplog-gig
Router(config-cmap)# match input-interface gigabitethernet 0/3
Router(config-cmap)# match packets dropped
Router(config-cmap)# exit

! Define a policy map of type logging using your logging type class map.
Router(config)# policy-map type logging cpplog-gig-policy

Router(config-pmap)# class cpplog-gig
Router(config-pmap-c)# log
Router(config-pmap-c)# exit
Router(config-pmap)# exit

! Apply the new logging policy map to the aggregate control-plane interface.
Router(config)#  control-plane
Router(config-cp)# service-policy type logging input cpplog-gig-policy
Router(config-cp)# end
Router#
Aug  8 12:53:08.618: %SYS-5-CONFIG_I: Configured from console by cisco on console
Router#
```

The following output displays the logging policy map that was just added to the aggregate control-plane interface:

```
Router# show policy-map type logging control-plane

Control Plane
    Service-policy logging input: cpplog-gig-policy
    Class-map: cpplog-gig (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: input-interface GigabitEthernet0/3
      Match:  dropped-packets
      log
    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
```

# Configuring Logging for a Specific Class Example

The following example shows how to configure class-specific control plane logging for a specific class configured in a control plane policing service policy. This example also shows how to configure rate-limiting of logs to output only one log message every 5 seconds. For this example, you have a control plane policing service policy with classes defined for Interior Gateway Protocol (IGP), management, monitoring and, undesirable traffic. The undesirable class is configured to match packets that are destined to the router on UDP port 1434. The service policy is configured to drop all packets that hit the undesirable class (in this case, packets that are destined for port 1434). For this example, you want to log all packets being dropped by the undesirable class, so that you will be aware that you are being attacked by 1434 packets.

In this example, you have the following control plane policing service policy configured:

```
Router# show policy-map cpp-policy

  Policy Map cpp-policy
    Class cppclass-igp
    Class cppclass-management
     police rate 250 pps burst 61 packets
       conform-action transmit
       exceed-action drop
    Class cppclass-monitoring
     police rate 100 pps burst 24 packets
       conform-action transmit
       exceed-action drop
    Class cppclass-undesirable
      drop
    Class class-default
     police rate 50 pps burst 12 packets
       conform-action transmit
       exceed-action drop
```

To log all traffic for the undesirable class in the above service policy, perform the following steps:

```
! Enter control plane policing policy-map configuration mode.
Router# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#  policy-map cpp-policy

! Enter policy-map class configuration mode for the undesirable class.
Router(config-pmap)# class cppclass-undesirable

! Configure the log keyword with a rate limit of one log message every 5 seconds.
Router(config-pmap-c)# log interval 5000
Router(config-pmap-c)# end
```

Use the following command to verify that the log action has been added to the policy map under the undesirable class:

```
Router# show policy-map cpp-policy

  Policy Map cpp-policy
    Class cppclass-igp
    Class cppclass-management
     police rate 250 pps burst 61 packets
       conform-action transmit
       exceed-action drop
    Class cppclass-monitoring
     police rate 100 pps burst 24 packets
 conform-action transmit
       exceed-action drop
Class cppclass-undesirable
      drop
log interval 5000
```

```
    Class class-default
     police rate 50 pps burst 12 packets
       conform-action transmit
       exceed-action drop
```

## Configuring Logging for a Port-Filter Policy Map Example

The following example shows how to configure class-specific control plane logging for a specific class configured in a Control Plane Protection port-filter policy map. This example also shows how to configure logging to display the packet-length field from the IP header for each packet that hits the port-filter class. For this example, you have a port-filter policy map configured to drop all traffic that is destined to closed TCP/UDP ports. For this example, you want to log all packets that are being dropped or allowed by the port-filter class.

In this example, you have the following port-filter service policy configured and applied to your control-plane host feature path. This policy blocks all traffic that is destined to closed or unlistened TCP/UDP ports:

```
Router# show policy-map type port-filter

  Policy Map type port-filter pf-closed-port-policy
    Class pf-closed-ports
      Drop
```

The corresponding port-filter type class map that is used in the above port-filter policy map is configured as follows:

```
Router# show class-map type port-filter

Class Map type port-filter match-all pf-closed-ports (id 19)
   Match  closed-ports
```

To log all traffic that is processed by the above pf-closed-ports class map in the above pf-closed-port-policy port-filter policy map, perform the following steps:

```
! Enter port-filter policy-map configuration mode.
Router# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# policy-map type port-filter pf-closed-port-policy

! Enter port-filter policy-map class configuration mode for the undesirable class.
Router(config-pmap)# class pf-closed-ports

! Configure the log keyword with the option to log the packet-length field in the IP header.

Router(config-pmap-c)# log total-length
Router(config-pmap-c)# end
```

Use the following command to verify that the log action has been added to the port-filter policy map under the appropriate class:

```
Router# show policy-map type port-filter
  Policy Map type port-filter pf-closed-port-policy
    Class pf-closed-ports
 drop
log interval 1000 total-length
```

# Additional References

The following sections provide references related to the Control Plane Logging feature.

### Related Documents

| Related Topic | Document Title |
|---|---|
| QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | Cisco IOS Quality of Service Solutions Command Reference |
| QoS feature overview | “Quality of Service Overview” module |
| Control plane protection | “Control Plane Protection” module |

### Standards

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

### MIBs

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

### RFCs

| RFC | Title |
|---|---|
| None | — |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content. | http://www.cisco.com/techsupport |

# Feature Information for Control Plane Logging

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 . An account on Cisco.com is not required.

***Table 7: Feature Information for Control Plane Logging***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Control Plane Logging | 12.4(6)T | Allows the control plane features to log all packets that match the class-map entries.<br>The following commands were introduced or modified: **class-map**, **debug control-plane** , **policy-map** |

CHAPTER 14

# Management Plane Protection

First Published: February 27, 2006

Last Updated: February 27, 2006

The Management Plane Protection (MPP) feature in Cisco IOS software provides the capability to restrict the interfaces on which network management packets are allowed to enter a device. The MPP feature allows a network operator to designate one or more router interfaces as management interfaces. Device management traffic is permitted to enter a device only through these management interfaces. After MPP is enabled, no interfaces except designated management interfaces will accept network management traffic destined to the device.

Restricting management packets to designated interfaces provides greater control over management of a device, providing more security for that device. Other benefits include improved performance for data packets on nonmanagement interfaces, support for network scalability, need for fewer access control lists (ACLs) to restrict access to a device, and management packet floods on switching and routing interfaces are prevented from reaching the CPU.

### Finding Feature Information in This Module

*Your Cisco IOS software release may not support all of the features documented in this module.* To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the Feature Information for Management Plane Protection, on page 156.

### Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn . You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

# Finding Feature Information

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

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

# Prerequisites for Management Plane Protection

- IP Cisco Express Forwarding must be enabled before a management interface can be configured.

# Restrictions for Management Plane Protection

- Out-of-band management interfaces (also called dedicated management interfaces) are not supported. An out-of-band management interface is a dedicated Cisco IOS physical or logical interface that processes management traffic only.
- Loopback and virtual interfaces not associated to physical interfaces are not supported.
- Fallback and standby management interfaces are not supported.
- Hardware-switched and distributed platforms are not supported.
- Secure Copy (SCP) is supported under the Secure Shell (SSH) Protocol and not directly configurable in the command-line interface (CLI).
- Uninformed management stations lose access to the router through nondesignated management interfaces when the Management Plane Protection feature is enabled.
- This feature supports only IPv4 traffic. IPv6 traffic is neither blocked nor denied.

# Information About Management Plane Protection

Before you enable the Management Plane Protection feature, you should understand the following concepts:

## In-Band Management Interface

An in-band management interface is a Cisco IOS physical or logical interface that processes management as well as data-forwarding packets. Loopback interfaces commonly are used as the primary port for network management packets. External applications communicating with a networking device direct network management requests to the loopback port. An in-band management interface is also called a shared management interface.

## Control Plane Protection Overview

A control plane is a collection of processes that run at the process level on a route processor and collectively provide high-level control for most Cisco IOS software functions. All traffic directly or indirectly destined to a router is handled by the control plane.

Control Plane Policing (CoPP) is a Cisco IOS control-plane feature that offers rate limiting of all control-plane traffic. CoPP allows you to configure a quality of service (QoS) filter that manages the traffic flow of control plane packets. This QoS filter helps to protect the control plane of Cisco IOS routers and switches against denial-of-service (DoS) attacks and helps to maintain packet forwarding and protocol states during an attack or during heavy traffic loads.

Control Plane Protection is a framework that encompasses all policing and protection features in the control plane. The Control Plane Protection feature extends the policing functionality of the CoPP feature by allowing finer policing granularity. Control Plane Protection also includes a traffic classifier, which intercepts control-plane traffic and classifies it in control-plane categories. Management Plane Protection operates within the Control Plane Protection infrastructure.

For more information about the Control Plane Policing feature in Cisco IOS software, see the Control Plane Policing module.

For more information about the Control Plane Protection feature in Cisco IOS software, see the Control Plane Protection module .

## Management Plane

The management plane is the logical path of all traffic related to the management of a routing platform. One of three planes in a communication architecture that is structured in layers and planes, the management plane performs management functions for a network and coordinates functions among all the planes (management, control, data). The management plane also is used to manage a device through its connection to the network.

Examples of protocols processed in the management plane are Simple Network Management Protocol (SNMP), Telnet, HTTP, Secure HTTP (HTTPS), and SSH. These management protocols are used for monitoring and for CLI access. Restricting access to devices to internal sources (trusted networks) is critical.

## Management Plane Protection Feature

The MPP feature in Cisco IOS software provides the capability to restrict the interfaces on which network management packets are allowed to enter a device. The MPP feature allows a network operator to designate one or more router interfaces as management interfaces. Device management traffic is permitted to enter a device through these management interfaces. After MPP is enabled, no interfaces except designated management interfaces will accept network management traffic destined to the device. Restricting management packets to designated interfaces provides greater control over management of a device.

The MPP feature is disabled by default. When you enable the feature, you must designate one or more interfaces as management interfaces and configure the management protocols that will be allowed on those interfaces. The feature does not provide a default management interface. Using a single CLI command, you can configure, modify, or delete a management interface.When you configure a management interface, no interfaces except that management interface will accept network management packets destined to the device. When the last configured interface is deleted, the feature turns itself off.

Following are the management protocols that the MPP feature supports. These management protocols are also the only protocols affected when MPP is enabled.

- Blocks Extensible Exchange Protocol (BEEP)
- FTP
- HTTP
- HTTPS
- SSH, v1 and v2
- SNMP, all versions
- Telnet
- TFTP

Cisco IOS features enabled on management interfaces remain available when the MPP feature is enabled. Nonmanagement packets such as routing and Address Resolution Protocol (ARP) messages for in-band management interfaces are not affected.

This feature generates a syslog for the following events:

- When the feature is enabled or disabled
- When a management interface fails.

For example, a failure will occur when the management interface cannot successfully receive or process packets destined for the control plane for reasons other than resource exhaustion.

## Benefits of the Management Plane Protection Feature

Implementing the MPP feature provides the following benefits:

- Greater access control for managing a device than allowing management protocols on all interfaces
- Improved performance for data packets on nonmanagement interfaces
- Support for network scalability
- Simplifies the task of using per-interface ACLs to restrict management access to the device
- Fewer ACLs needed to restrict access to the device
- Management packet floods on switching and routing interfaces are prevented from reaching the CPU

# How to Configure a Device for Management Plane Protection

This section contains the following task:

# Configuring a Device for Management Plane Protection

Perform this task to configure a device that you have just added to your network or a device already operating in your network. This task shows how to configure MPP where SSH and SNMP are allowed to access the router only through the FastEthernet 0/0 interface.

### Before You Begin

- IP Cisco Express Forwarding must be enabled before a management interface can be configured.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane host**
4. **management-interface** *interface* **allow** *protocols*
5. **Ctrl z**
6. **show management-interface** [*interface* | **protocol** *protocol-name*]

## 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** | **control-plane host**<br>**Example:**<br>`Router(config)# control-plane host` | Enters control-plane host configuration mode. |
| **Step 4** | **management-interface** *interface* **allow** *protocols*<br>**Example:**<br>`Router(config-cp-host)# management-interface FastEthernet 0/0 allow ssh snmp` | Configures an interface to be a management interface, which will accept management protocols, and specifies which management protocols are allowed.<br>• *interface*—Name of the interface that you are designating as a management interface.<br>**Note** Effective with Cisco IOS XE Release 3.16S, you can configure a virtual template interface. |

| | Command or Action | Purpose |
|---|---|---|
| | | • *protocols*—Management protocols you want to allow on the designated management interface.<br>◦ BEEP<br>◦ FTP<br>◦ HTTP<br>◦ HTTPS<br>◦ SSH, v1 and v2<br>◦ SNMP, all versions<br>◦ Telnet<br>◦ TFTP |
| **Step 5** | **Ctrl z**<br>**Example:**<br>`Router(config-cp-host)# Ctrl z` | Returns to privileged EXEC mode. |
| **Step 6** | **show management-interface** [*interface* \| **protocol** *protocol-name*]<br>**Example:**<br>`Router# show management-interface FastEthernet 0/0` | Displays information about the management interface such as type of interface, protocols enabled on the interface, and number of packets dropped and processed.<br>*interface*—(Optional) Interface for which you want to view information.<br>**protocol**—(Optional) Indicates that a protocol is specified.<br>*protocol-name*—(Optional) Protocol for which you want to view information |

## Examples

The configuration in this example shows MPP configured to allow SSH and SNMP to access the router only through the FastEthernet 0/0 interface. This configuration results in all protocols in the remaining subset of supported management protocols to be dropped on all interfaces unless explicitly permitted. BEEP, FTP, HTTP, HTTPS, Telnet, and TFTP will not be permitted to access the router through any interfaces, including FastEthernet 0/0. Additionally, SNMP and SSH will be dropped on all interfaces except FastEthernet 0/0, where it is explicitly allowed.

To allow other supported management protocols to access the router, you must explicitly allow these protocols by adding them to the protocol list for the FastEthernet 0/0 interface or enabling additional management interfaces and protocols.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# control-plane host
Router(config-cp-host)# management-interface FastEthernet 0/0 allow ssh snmp
Router(config-cp-host)#
.Aug  2 15:25:32.846: %CP-5-FEATURE: Management-Interface feature enabled on Control plane
 host path
Router(config-cp-host)#
```

The following is output from the **show management-interface** command issued after configuring MPP in the previous example. The **show management-interface** command is useful for verifying your configuration.

```
Router# show management-interface
Management interface FastEthernet0/0
        Protocol        Packets processed
             ssh                0
            snmp                0
Router#
```

# Configuration Examples for Management Plane Protection

This section provides the following configuration example:

## Configuring Management Plane Protection on Gigabit Ethernet Interfaces: Example

The following example shows how to configure MPP where only SSH, SNMP, and HTTP are allowed to access the router through the Gigabit Ethernet 0/3 interface and only HTTP is allowed to access the router through the Gigabit Ethernet 0/2 interface.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# control-plane host
Router(config-cp-host)# management-interface GigabitEthernet 0/3 allow http ssh snmp

Router(config-cp-host)#
.Aug  2 17:00:24.511: %CP-5-FEATURE: Management-Interface feature enabled on Control plane
 host path
Router(config-cp-host)# management-interface GigabitEthernet 0/2 allow http
Router(config-cp-host)#
```

The following is output from the **show management-interface** command issued after configuring MPP in the previous example. The **show management-interface** command is useful for verifying your configuration.

```
Router# show management-interface

Management interface GigabitEthernet0/2
        Protocol        Packets processed
            http                0
Management interface GigabitEthernet0/3
        Protocol        Packets processed
            http                0
             ssh                0
            snmp                0
```

# Additional References for Management Plane Protection

The following sections provide references related to Management Plane Protection.

### Related Documents

| Related Topic | Document Title |
|---|---|
| Network management | Cisco IOS Network Management Configuration Guide |
| Network security | Cisco IOS Security Configuration Guide |
| Control Plane Policing | Control Plane Policing module |
| Control Plane Protection | Control Plane Protection module |

### RFCs

| RFC | Title |
|---|---|
| RFC 3871 | Operational Security Requirements for Large Internet Service Provider (ISP) IP Network Infrastructure |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Technical Support and Documentation website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content. | http://www.cisco.com/techsupport |

# Feature Information for Management Plane Protection

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 . An account on Cisco.com is not required.

*Table 8: Feature Information for Management Plane Protection*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Management Plane Protection | 12.4(6)T | Provides the capability to restrict the interfaces on which network management packets are allowed to enter a device. |

CHAPTER 15

# Class-Based Policing

### Feature History

| Release | Modification |
|---|---|
| 12.1(5)T | This command was introduced for Cisco IOS Release 12.1 T. A new Class-Based Policing algorithm was introduced. The **violate-action** option became available. This feature became available on Cisco 2600, 3600, 4500, 7200, and 7500 series routers. |
| 12.2(2)T | The set-clp-transmit option for the *action* argument was added to the **police** command. The set-frde-transmit option for the *action* argument was added to the **police** command. The set-mpls-exp-transmit option for the *action* argument was added to the **police** command. |
| 12.0(26)S | This feature was integrated into Cisco IOS Release 12.0(26)S for the Cisco 7200 and 7500 series routers. The name of the feature changed from *Traffic Policing* to *Class-Based Policing*. |

# 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 at the end of this module.

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.

# Feature Overview

Class-Based Policing allows you to control the maximum rate of traffic transmitted or received on an interface. The Class-Based Policing feature is applied when you attach a traffic policy contain the Class-Based Policing configuration to an interface.

The Class-Based Policing feature performs the following functions:

- Limits the input or output transmission rate of a class of traffic based on user-defined criteria
- Marks packets by setting the ATM Cell Loss Priority (CLP) bit, Frame Relay Discard Eligibility (DE) bit, IP precedence value, IP differentiated services code point (DSCP) value, MPLS experimental value, and Quality of Service (QoS) group.

## Benefits

### Bandwidth Management Through Rate Limiting

Class-Based Policing allows you to control the maximum rate of traffic transmitted or received on an interface. Class-Based Policing is often configured on interfaces at the edge of a network to limit traffic into or out of the network. In most Class-Based Policing configurations, traffic that falls within the rate parameters is transmitted, whereas traffic that exceeds the parameters is dropped or transmitted with a different priority.

### Packet Marking

Packet marking allows you to partition your network into multiple priority levels or classes of service (CoS). A packet is marked and these markings can be used to identify and classify traffic for downstream devices. In some cases, such as ATM Cell Loss Priority (CLP) marking or Frame Relay Discard Eligibility (DE) marking, the marking is used to classify traffic.

- Use Class-Based Policing to set the IP precedence or DSCP values for packets entering the network. Networking devices within your network can then use the adjusted IP precedence values to determine how the traffic should be treated. For example, the Weighted Random Early Detection (WRED) feature uses the IP precedence values to determine the probability that a packet will be dropped.
- Use Class-Based Policing to assign packets to a QoS group. The router uses the QoS group to determine how to prioritize packets within the router.

Traffic can be marked without using the Class-Based Policing feature. If you want to mark traffic but do not want to use Class-Based Policing, see the "Marking Network Traffic" module.

### Packet Prioritization for Frame Relay Frames

The Class-Based Policing feature allows users to mark the Frame Relay DE bit of the Frame Relay frame. The Frame Relay DE bit is one bit and, therefore, can be set to either 0 or 1. In congested environments, frames with the DE bit set to 1 are discarded before frames with the DE bit set to 0.

### Packet Prioritization for ATM Cells

The Class-Based Policing feature allows users to mark the ATM CLP bit in ATM cells. The ATM CLP bit is used to prioritize packets in ATM networks. The ATM CLP bit is one bit and, therefore, can be set to either 0 or 1. In congested environments, cells with the ATM CLP bit set to 1 are discarded before cells with the ATM CLP bit set to 0.

# Restrictions

- To use the *set-clp-transmit*action available with this feature, the Enhanced ATM Port Adapter (PA-A3) is required. Therefore, the *set-clp-transmit* action is not supported on any platform that does not support the PA-A3 adapter (such as the Cisco 2600 series router, the Cisco 3640 router, and the 4500 series router). For more information, see the documentation for your specific device.
- On a Cisco 7500 series router, Class-Based Policing can monitor Cisco Express Forwarding (CEF) switching paths only. In order to use the Class-Based Policing feature, Cisco Express Forwarding must be configured on both the interface receiving the packet and the interface sending the packet.
- On a Cisco 7500 series router, Class-Based Policing cannot be applied to packets that originated from or are destined to a device.
- Class-Based Policing can be configured on an interface or a subinterface.
- Class-Based Policing is not supported on the following interfaces:
    - Fast EtherChannel
    - PRI
    - Any interface on a Cisco 7500 series router that does not support Cisco Express Forwarding
    - Tunnel

**Note** Class-Based Policing is supported on tunnels that are using the Cisco generic routing encapsulation (GRE) tunneling protocol.

# Prerequisites

On a Cisco 7500 series router, Cisco Express Forwarding (CEF) must be configured on the interface before Class-Based Policing can be used.

# Configuration Tasks

## Configuring Traffic Policing

| Command | Purpose |
|---|---|
| Device(config-pmap-c)# **police** *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action* **violate-action** *action* | Specifies a maximum bandwidth usage by a traffic class.<br>**Note** The Class-Based Policing feature works with a token bucket mechanism. There are currently two types of token bucket algorithms: a single token bucket algorithm and a two token bucket algorithm. A single token bucket system is used when the **violate-action** option is not specified, and a two token bucket system is used when the **violate-action** option is specified. |

## Verifying Traffic Policing

Use the **show policy-map interface** EXEC command to verify that the Class-Based Policing feature is configured on your interface. If the feature is configured on your interface, the **show policy-map interface** command output displays policing statistics:

```
Device# show policy-map interface
 Ethernet1/7
  service-policy output: x
    class-map: a (match-all)
      0 packets, 0 bytes
      5 minute rate 0 bps
      match: ip precedence 0
      police:
        1000000 bps, 10000 limit, 10000 extended limit
        conformed 0 packets, 0 bytes; action: transmit
        exceeded 0 packets, 0 bytes; action: drop
        conformed 0 bps, exceed 0 bps, violate 0 bps
```

## Troubleshooting Tips

- Check the interface type. Verify that your interface is not mentioned in the nonsupported interface description in the Restrictions, on page 161 section of this module.
- For input Class-Based Policing on a Cisco 7500 series router, verify that CEF is configured on the interface where Class-Based Policing is configured.
- For output Class-Based Policing on a Cisco 7500 series router, ensure that the incoming traffic is CEF-switched. Class-Based Policing cannot be used on the switching path unless CEF switching is enabled.

# Monitoring and Maintaining Traffic Policing

| Command | Purpose |
|---|---|
| Device# **show policy-map** | Displays all configured policy maps. |
| Device# **show policy-map** *policy-map-name* | Displays the user-specified policy map. |
| Device# **show policy-map interface** | Displays statistics and configurations of all input and output policies that are attached to an interface. |
| Device# **show policy-map interface service instance** | Displays the policy map information for a given service instance under a port channel. |

# Configuration Examples

## Example Configuring a Service Policy that Includes Traffic Policing

In the following example, Class-Based Policing is configured with the average rate at 8000 bits per second, the normal burst size at 1000 bytes, and the excess burst size at 1000 bytes for all packets leaving Fast Ethernet interface 0/0.

```
class-map access-match
match access-group 1
exit
policy-map police-setting
class access-match
police 8000 1000 1000 conform-action transmit exceed-action set-qos-transmit 1 violate-action
 drop
exit
exit
service-policy output police-setting
```

The treatment of a series of packets leaving Fast Ethernet interface 0/0 depends on the size of the packet and the number of bytes remaining in the conform and exceed token buckets. The series of packets are policed based on the following rules:

- If the previous arrival of the packet was at T1 and the current arrival of the packet is at T, the bucket is updated with T -T1 worth of bits based on the token arrival rate. The refill tokens are placed in the conform bucket. If the tokens overflow the conform bucket, the overflow tokens are placed in the exceed bucket. The token arrival rate is calculated as follows:

(time between packets <which is equal to T - T1> * policer rate)/8 bytes

- If the number of bytes in the conform bucket B is greater than or equal to 0, the packet conforms and the conform action is taken on the packet. If the packet conforms, B bytes are removed from the conform bucket and the conform action is taken. The exceed bucket is unaffected in this scenario.
- If the number of bytes in the conform bucket B is less than 0, the excess token bucket is checked for bytes by the packet. If the number of bytes in the exceed bucket B is greater than or equal to 0, the exceed

action is taken and B bytes are removed from the exceed token bucket. No bytes are removed from the conform bucket in this scenario.

- If the number bytes in the exceed bucket B is fewer than 0, the packet violates the rate and the violate action is taken. The action is complete for the packet.

In this example, the initial token buckets starts full at 1000 bytes. If a 450-byte packet arrives, the packet conforms because enough bytes are available in the conform token bucket. The conform action (send) is taken by the packet and 450 bytes are removed from the conform token bucket (leaving 550 bytes).

If the next packet arrives 0.25 seconds later, 250 bytes are added to the conform token bucket

((0.25 * 8000)/8), leaving 800 bytes in the conform token bucket. If the next packet is 900 bytes, the packet does not conform because only 800 bytes are available in the conform token bucket.

The exceed token bucket, which starts full at 1000 bytes (as specified by the excess burst size) is then checked for available bytes. Because enough bytes are available in the exceed token bucket, the exceed action (set the QoS transmit value of 1) is taken and 900 bytes are taken from the exceed bucket (leaving 100 bytes in the exceed token bucket.

If the next packet arrives 0.40 seconds later, 400 bytes are added to the token buckets ((.40 * 8000)/8). Therefore, the conform token bucket now has 1000 bytes (the maximum number of tokens available in the conform bucket) and 200 bytes overflow the conform token bucket (because it only 200 bytes were needed to fill the conform token bucket to capacity). These overflow bytes are placed in the exceed token bucket, giving the exceed token bucket 300 bytes.

If the arriving packet is 1000 bytes, the packet conforms because enough bytes are available in the conform token bucket. The conform action (transmit) is taken by the packet, and 1000 bytes are removed from the conform token bucket (leaving 0 bytes).

If the next packet arrives 0.20 seconds later, 200 bytes are added to the token bucket ((.20 * 8000)/8). Therefore, the conform bucket now has 200 bytes. If the arriving packet is 400 bytes, the packet does not conform because only 200 bytes are available in the conform bucket. Similarly, the packet does not exceed because only 300 bytes are available in the exceed bucket. Therefore, the packet violates and the violate action (drop) is taken.

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | *Cisco IOS Quality of Service Solutions Command Reference* |
| Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC), hierarchical policies, policy maps | "Applying QoS Features Using the MQC" module |
| Policing and shaping traffic | "Policing and Shaping Overview" module |

### Standards

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

### MIBs

| MIB | MIBs Link |
|---|---|
| No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature. | To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL:<br>http://www.cisco.com/go/mibs |

### RFCs

| RFC | Title |
|---|---|
| No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature. | -- |

### Technical Assistance

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

CHAPTER 16

# QoS Percentage-Based Policing

The QoS Percentage-Based Policing feature allows you to configure traffic policing and traffic shaping on the basis of a percentage of bandwidth available on the interface. This feature also allows you to specify the committed burst (bc) size and the excess burst (be) size (used for configuring traffic policing) in milliseconds (ms). Configuring traffic policing in this manner enables you to use the same policy map for multiple interfaces with differing amounts of bandwidth.

## 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 at the end of this module.

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

## Prerequisites for QoS Percentage-Based Policing

- For input traffic policing on a Cisco 7500 series router, verify that distributed Cisco Express Forwarding (dCEF) is enabled on the interface on which traffic policing is configured.

- For output traffic policing on a Cisco 7500 series router, ensure that the incoming traffic is dCEF-switched. Traffic policing cannot be used on the switching path unless dCEF switching is enabled.

# Restrictions for QoS Percentage-Based Policing

The **shape** (percent) command, when used in "child" (nested) policy maps, is not supported on the Cisco 7500, the Cisco 7200, or lower series routers. Therefore, the **shape** (percent) command cannot be configured for use in nested policy maps on these routers.

# Information About QoS Percentage-Based Policing

## Benefits for QoS Percentage-Based Policing

This feature provides the ability to configure traffic policing and traffic shaping on the basis of a percentage of bandwidth available on an interface, and it allows you to specify burst sizes in milliseconds. Configuring traffic policing and traffic shaping in this manner enables you to use the same policy map for multiple interfaces with differing amounts of bandwidth. That is, you do not have to recalculate the bandwidth for each interface or configure a different policy map for each type of interface.

## Defining Class and Policy Maps for QoS Percentage-Based Policing

To configure the QoS Percentage-Based Policing feature, you must define a traffic class, configure a policy map, and then attach that policy map to the appropriate interface. These three tasks can be accomplished by using the Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC).

The MQC is a command-line interface that allows you to define traffic classes, create and configure traffic policies (policy maps), and then attach these traffic policies to interfaces.

In the MQC, the **class-map** command is used to define a traffic class (which is then associated with a traffic policy). The purpose of a traffic class is to classify traffic.

The MQC consists of the following three processes:

- Defining a traffic class with the **class-map** command.
- Creating a traffic policy by associating the traffic class with one or more QoS features (using the **policy-map** command).
- Attaching the traffic policy to the interface with the **service-policy** command.

A traffic class contains three major elements: a name, a series of match commands, and, if more than one **match** command exists in the traffic class, an instruction on how to evaluate these **match** commands (that is, match-all or match-any). The traffic class is named in the **class-map** command line; for example, if you enter the **class-map cisco** command while configuring the traffic class in the CLI, the traffic class would be named "cisco".

The **match** commands are used to specify various criteria for classifying packets. Packets are checked to determine whether they match the criteria specified in the **match** commands. If a packet matches the specified criteria, that packet is considered a member of the class and is forwarded according to the QoS specifications set in the traffic policy. Packets that fail to meet any of the matching criteria are classified as members of the default traffic class.

# Traffic Regulation Mechanisms and Bandwidth Percentages

Cisco IOS quality of service (QoS) offers two kinds of traffic regulation mechanisms--traffic policing and traffic shaping. A traffic policer typically drops traffic that violates a specific rate. A traffic shaper typically delays excess traffic using a buffer to hold packets and shapes the flow when the data rate to a queue is higher than expected.

Traffic shaping and traffic policing can work in tandem and can be configured in a class map. Class maps organize data packets into specific categories ("classes") that can, in turn, receive a user-defined QoS treatment when used in policy maps (sometimes referred to as "service policies").

Before this feature, traffic policing and traffic shaping were configured on the basis of a user-specified amount of bandwidth available on the interface. Policy maps were then configured on the basis of that specific amount of bandwidth, meaning that separate policy maps were required for each interface.

This feature provides the ability to configure traffic policing and traffic shaping on the basis of a *percentage* of bandwidth available on the interface. Configuring traffic policing and traffic shaping in this manner enables customers to use the same policy map for multiple interfaces with differing amounts of bandwidth.

Configuring traffic policing and shaping on the basis of a percentage of bandwidth is accomplished by using the **police** (percent) and **shape** (percent) commands.

# Burst Size in Milliseconds Option

The purpose of the burst parameters (bc and be) is to drop packets gradually, as is done with Weighted Random Early Detection (WRED), and to avoid tail drop. Setting sufficiently high burst values helps to ensure good throughput.

This feature allows you the option of specifying the committed burst (bc) size and the extended burst (be) as milliseconds (ms) of the class bandwidth when you configure traffic policing. The number of milliseconds is used to calculate the number of bytes that will be used by the QoS Percentage-Based Policing feature.

Specifying these burst sizes in milliseconds is accomplished by using the **bc** and **be** keywords (and their associated arguments) of the **police** (percent) and **shape** (percent) commands.

# How to Configure QoS Percentage-Based Policing

## Configuring a Class and Policy Map for Percentage-Based Policing

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-name*
4. **class** {*class-name* **class-default**}
5. **police cir percent** *percentage* [*burst-in-ms*] [**bc** *conform-burst-in-msec* **ms**] [**be** *peak-burst-in-msec* **ms**] [ **pir percent** *percent*]
6. **exit**

### 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** *policy-name*<br>**Example:**<br>`Router(config)# policy-map`<br>`policy1` | Specifies the name of the policy map to be created. Enters policy-map configuration mode.<br>• Enter the policy map name. |
| **Step 4** | **class** {*class-name* **class-default**}<br>**Example:**<br>`Router(config-pmap)# class class1` | Specifies the class so that you can configure or modify its policy. Enters policy-map class configuration mode.<br>• Enter the class name or specify the default class (class-default). |
| **Step 5** | **police cir percent** *percentage* [*burst-in-ms*] [**bc** *conform-burst-in-msec* **ms**] [**be** *peak-burst-in-msec* **ms**] [ **pir percent** *percent*] | Configures traffic policing on the basis of the specified bandwidth percentage and optional burst sizes. Enters policy-map class police configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config-pmap-c)# police cir percent 20 bc 300 ms be 400 ms pir percent 40` | • Enter the bandwidth percentage and optional burst sizes. |
| **Step 6** | **exit**<br>**Example:**<br>`Router(config-pmap-c-police)# exit` | Exits policy-map class police configuration mode. |

# Attaching the Policy Map to an Interface for Percentage-Based Policing

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **pvc** [*name*] *vpi* / *vci* [**ilmi** | **qsaal** | **smds**]
5. **service-policy** {**input**| **output**} *policy-map-name*
6. **exit**

## 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** | **interface** *type number*<br>**Example:**<br>`Router(config)#` | Configures an interface (or subinterface) type and enters interface configuration mode.<br>• Enter the interface type number. |

| | Command or Action | Purpose |
|---|---|---|
| | `interface serial4/0` | **Note** Depending on the needs of your network, you may need to attach the policy map to a subinterface, an ATM PVC, a Frame Relay DLCI, or other type of interface. |
| **Step 4** | **pvc** [*name*] *vpi* / *vci* [**ilmi** \| **qsaal** \| **smds**]<br>**Example:**<br>`Router(config-if)# pvc cisco 0/16 ilmi` | (Optional) Creates or assigns a name to an ATM PVC and specifies the encapsulation type on an ATM PVC. Enters ATM VC configuration mode.<br>**Note** This step is required only if you are attaching the policy map to an ATM PVC. If you are not attaching the policy map to an ATM PVC, skip this step and proceed with Attaching the Policy Map to an Interface for Percentage-Based Policing. |
| **Step 5** | **service-policy** {**input**\| **output**} *policy-map-name*<br>**Example:**<br>`Router(config-if)#`<br>`service-policy input policy1`<br>**Example:** | Specifies the name of the policy map to be attached to the input *or* output direction of the interface.<br>**Note** Policy maps can be configured on ingress or egress routers. They can also be attached in the input or output direction of an interface. The direction (input or output) and the router (ingress or egress) to which the policy map should be attached varies according your network configuration. When using the **service-policy** command to attach the policy map to an interface, be sure to choose the router and the interface direction that are appropriate for your network configuration.<br>• Enter the policy map name. |
| **Step 6** | **exit**<br>**Example:**<br>`Router(config-if)# exit` | (Optional) Exits interface configuration mode. |

# Verifying the Percentage-Based Policing Configuration

## SUMMARY STEPS

1. **enable**
2. **show class-map** [*class-map-name*]
3. **show policy-map interface** *interface-name*
4. **exit**

## 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** | **show class-map** [*class-map-name*]<br>**Example:**<br>`Router# show class-map class1` | Displays all information about a class map, including the match criterion.<br>• Enter class map name. |
| **Step 3** | **show policy-map interface** *interface-name*<br>**Example:**<br>`Router#`<br>`show policy-map interface serial4/0` | Displays the packet statistics of all classes that are configured for all service policies either on the specified interface or subinterface or on a specific PVC on the interface.<br>• Enter the interface name. |
| **Step 4** | **exit**<br>**Example:**<br>`Router# exit` | (Optional) Exits privileged EXEC mode. |

## Troubleshooting Tips for Percentage-Based Policing

The commands in the Verifying the Percentage-Based Policing Configuration, on page 172 section allow you to verify that you achieved the intended configuration and that the feature is functioning correctly. If, after using the **show** commands listed above, you find that the configuration is not correct or the feature is not functioning as expected, perform these operations:

If the configuration is not the one you intended, complete the following procedures:

1. Use the **show running-config** command and analyze the output of the command.
2. If the policy map does not appear in the output of the **show running-config** command, enable the **logging console** command.
3. Attach the policy map to the interface again.

If the packets are not being matched correctly (for example, the packet counters are not incrementing correctly), complete the following procedures:

1. Run the **show policy-map**command and analyze the output of the command.
2. Run the **show running-config** command and analyze the output of the command.
3. Use the **show policy-map interface** command and analyze the output of the command. Check the the following findings:

1. If a policy map applies queueing, and the packets are matching the correct class, but you see unexpected results, compare the number of the packets in the queue with the number of the packets matched.
2. If the interface is congested, and only a small number of the packets are being matched, check the tuning of the transmission (tx) ring, and evaluate whether the queueing is happening on the tx ring. To do this, use the **show controllers** command, and look at the value of the tx count in the output of the command.

# Configuration Examples for QoS Percentage-Based Policing

## Specifying Traffic Policing on the Basis of a Bandwidth Percentage Example

The following example configures traffic policing using a committed information rate (CIR) and a peak information rate (PIR) on the basis of a percentage of bandwidth. In this example, a CIR of 20 percent and a PIR of 40 percent have been specified. Additionally, an optional bc value and be value (300 ms and 400 ms, respectively) have been specified.

```
Router> enable
Router# configure terminal
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# police cir percent 20 bc 300 ms be 400 ms pir percent 40

Router(config-pmap-c-police)# exit
```

After the policy map and class maps are configured, the policy map is attached to interface as shown in the following example.

```
Router> enable
Router# configure terminal
Router(config-if)#

interface serial4/0
Router(config-if)#

service-policy input policy1
Router(config-if)# exit
```

## Verifying the Percentage-Based Policing Configuration Example

This section contains sample output from the **show policy-map interface** command and the **show policy-map** command. The output from these commands can be used to verify and monitor the feature configuration on your network.

The following is sample output from the **show policy-map** command. This sample output displays the contents of a policy map called "policy1." In policy 1, traffic policing on the basis of a CIR of 20 percent has been configured, and the bc and be have been specified in milliseconds. As part of the traffic policing configuration, optional conform, exceed, and violate actions have been specified.

```
Router# show policy-map policy1
  Policy Map policy1
    Class class1
     police cir percent 20 bc 300 ms pir percent 40 be 400 ms
       conform-action transmit
       exceed-action drop
       violate-action drop
```

The following is sample output from the **show policy-map interface** command. This sample displays the statistics for the serial 2/0 interface on which traffic policing has been enabled. The committed burst (bc) and excess burst (be) are specified in milliseconds (ms).

```
Router# show policy-map interface serial2/0
 Serial2/0
  Service-policy output: policy1 (1050)
    Class-map: class1 (match-all) (1051/1)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: ip precedence 0  (1052)
      police:
          cir 20 % bc 300 ms
          cir 409500 bps, bc 15360 bytes
          pir 40 % be 400 ms
          pir 819000 bps, be 40960 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        violated 0 packets, 0 bytes; actions:
          drop
        conformed 0 bps, exceed 0 bps, violate 0 bps
    Class-map: class-default (match-any) (1054/0)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any  (1055)
        0 packets, 0 bytes
        5 minute rate 0 bps
```

In this example, the CIR and PIR are displayed in bps, and both the committed burst (bc) and excess burst (be) are displayed in bits.

The CIR, PIR bc, and be are calculated on the basis of the formulas described below.

### Formula for Calculating the CIR

When calculating the CIR, the following formula is used:

- CIR percentage specified (as shown in the output of the **show policy-map**command) * bandwidth (BW) of the interface (as shown in the output of the **show interfaces** command) = total bits per second

On serial interface 2/0, the bandwidth (BW) is 2048 kbps. To see the bandwidth of the interface, use the **show interfaces** command. A sample is shown below:

```
Router# show interfaces serial2/0
Serial2/0 is administratively down, line protocol is down
  Hardware is M4T
  MTU 1500 bytes, BW 2048 Kbit, DLY 20000 usec, rely 255/255, load 1/255
```

The following values are used for calculating the CI:.

20 % * 2048 kbps = 409600 bps

### Formula for Calculating the PIR

When calculating the PIR, the following formula is used:

- PIR percentage specified (as shown in the output of the **show policy-map**command) * bandwidth (BW) of the interface (as shown in the output of the **show interfaces** command) = total bits per second

On serial interface 2/0, the bandwidth (BW) is 2048 kbps. To see the bandwidth of the interface, use the **show interfaces** command. A sample is shown below:

```
Router# show interfaces serial2/0
Serial2/0 is administratively down, line protocol is down
  Hardware is M4T
  MTU 1500 bytes, BW 2048 Kbit, DLY 20000 usec, rely 255/255, load 1/255
```

The following values are used for calculating the PIR:

40 % * 2048 kbps = 819200 bps

**Note** Discrepancies between this total and the total shown in the output of the **show policy-map interface** command can be attributed to a rounding calculation or to differences associated with the specific interface configuration.

### Formula for Calculating the Committed Burst (bc)

When calculating the bc, the following formula is used:

- The bc in milliseconds (as shown in the **show policy-map** command) * the CIR in bits per seconds = total number bytes

The following values are used for calculating the bc:

300 ms * 409600 bps = 15360 bytes

### Formula for Calculating the Excess Burst (be)

When calculating the bc and the be, the following formula is used:

- The be in milliseconds (as shown in the **show policy-map** command) * the PIR in bits per seconds = total number bytes

The following values are used for calculating the be:

400 ms * 819200 bps = 40960 bytes

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | *Cisco IOS Quality of Service Solutions Command Reference* |
| Modular QoS Command-Line Interface (CLI) (MQC), including information about attaching policy maps | "Applying QoS Features Using the MQC" module |
| Traffic shaping and traffic policing | "Policing and Shaping Overview" module |

### Standards

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

### MIBs

| MIB | MIBs Link |
|---|---|
| No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature. | To locate and download MIBs for selected platforms, Cisco IOS XE Software releases, and feature sets, use Cisco MIB Locator found at the following URL:<br>http://www.cisco.com/go/mibs |

### RFCs

| RFC | Title |
|---|---|
| RFC 2697 | *A Single Rate Three Color Marker* |
| RFC 2698 | *A Two Rate Three Color Marker* |

### 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 QoS Percentage-Based Policing

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 . An account on Cisco.com is not required.

***Table 9: Feature Information for QoS Percentage-Based Policing***

| Feature Name | Releases | Feature Information |
|---|---|---|
| QoS Percentage-Based Policing | 12.2(13)T 12.0(28)S 12.2(28)SB 15.0(1)S | The QoS Percentage-Based Policing feature allows you to configure traffic policing and traffic shaping on the basis of a *percentage* of bandwidth available on the interface. This feature also allows you to specify the committed burst (bc) size and the excess burst (be) size (used for configuring traffic policing) in milliseconds (ms). Configuring traffic policing in this manner enables you to use the same policy map for multiple interfaces with differing amounts of bandwidth.<br>In Release 12.2(13)T, this feature was introduced.<br>In Release 12.0(28)S, the option of specifying committed (conform) burst (bc) and excess (peak) burst (be) sizes in milliseconds was added.<br>In Release 12.2(28)SB, this feature was integrated in Cisco IOS Release 12.2(28)SB.<br>In Release 15.0(1)S, this feature was integrated in Cisco IOS Release 15.0(1)S.<br>The following commands were introduced or modified: **police (percent)**, **shape (percent)**, **show policy-map**, **show policy-map interface**. |

CHAPTER 17

# Overhead Accounting

Overhead accounting enables the router to account for packet overhead when shaping traffic to a specific rate. This accounting ensures that the router executes quality of service (QoS) features on the actual bandwidth used by subscriber traffic.

**Note** Overhead Accounting is not the same as Traffic Shaping Overhead Accounting for ATM, documented here: http://www.cisco.com/en/US/docs/ios/12_2sb/feature/guide/ovrhactg.html

## 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 at the end of this module.

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

## Restrictions for Overhead Accounting

- Overhead Accounting values are in bytes.

- You can set one Overhead Accounting value per policy.
- Overhead Accounting is supported only for the shape and bandwidth commands.
- Overhead Accountingis supported only on LAN and WAN interfaces.
- You can enable overhead accounting for shaping and bandwidth on top-level parent policies, middle-level child policies, and bottom-level child policies.
- When you enter the show policy-map interface command, the resulting classification byte counts and the queuing feature byte counts do not match. This mismatch occurs because the classification byte count does not consider overhead, whereas the queuing features do consider overhead.

# Information About Overhead Accounting

Overhead Accounting factors in packet datagram sizes. The following features use packet datagram size to make decisions in data plane operation; they use Overhead Accounting:

- Rate-limited priority queue (conditional policer)
- WRED (random drop)
- Qlimit (tail drop)
- Shaping
- Bandwidth operations

# How to Use Overhead Accounting

Overhead Accounting is disabled by default. You set the Overhead Accounting value. Configuring Overhead Accounting on a queue does not change queueing parameters you have already configured.

# Enabling Overhead Accounting

## SUMMARY STEPS

1. **enable**
2. **configure** {**terminal** | **memory** | **network**}
3. **policy-map** *policy-name*
4. **class**{*class-name* | **class-default**}
5. **bandwidth**{*bandwidth-kbps* |**percent***percentage*|**remaining percent***percentage*}[**account**{*subscriber-encap*} | {**user-defined***offset*}]
6. **exit**
7. **policy-map** *policy-name*
8. **class**{*class-name* | **class-default**}
9. **shape**[**average**|**peak**]*mean-rate* [*burst-size*] [*excess-burst-size*]**account**{*subscriber-encapsulation* |**user-defined***offset*}
10. **service-policy***policy-map-name*
11. **exit**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables higher privilege levels, such as privileged EXEC mode.<br>Enter your password if prompted. |
| **Step 2** | **configure** {**terminal** \| **memory** \| **network**}<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **policy-map** *policy-name*<br>**Example:**<br>`Router(config)# policy-map policy1` | Specifies the name of the policy map created earlier.<br>Enter policy map name. |
| **Step 4** | **class**{*class-name* \| **class-default**}<br>**Example:**<br>`Router(config-pmap)# class class1` | Enters policy-map configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | **bandwidth**{*bandwidth-kbps* \|**percent***percentage*\|**remaining percent***percentage*}[**account**{*subscriber-encap*} \| {**user-defined***offset*}]<br>**Example:**<br>`Router(config-pmap-c)# bandwidth percent 20` | Enables class-based fair queuing and overhead accounting. |
| **Step 6** | **exit**<br>**Example:**<br>`Router(config-pmap-c)# exit` | Exits the policy-map class configuration mode. |
| **Step 7** | **policy-map** *policy-name*<br>**Example:**<br>`Router(config)# policy-map policy1` | Specifies the name of the policy map created earlier.<br>Enter policy map name. |
| **Step 8** | **class**{*class-name* \| **class-default**}<br>**Example:**<br>`Router(config-pmap)# class class1` | Enters policy-map configuration mode. |
| **Step 9** | **shape**[**average**\|**peak**]*mean-rate* [*burst-size*] [*excess-burst-size*]**account**{*subscriber-encapsulation* \|**user-defined***offset*}<br>**Example:**<br>`Router(config-pmap-c)# shape average 1000000 account user-defined 33` | Shapes traffic to the indicated bit rate according to the algorithm specified or to enable overhead accounting. |
| **Step 10** | **service-policy***policy-map-name*<br>**Example:**<br>`Router(config-pmap-c)# service-policy oh-child-bw` | Applies a child policy to the parent class-default class. Do not specify the input or output keywords when applying a child policy to a parent class-default class. |
| **Step 11** | **exit**<br>**Example:**<br>`Router(config-pmap-c)# exit` | Exits the policy-map class configuration mode. |

# Verifying Overhead Accounting

## SUMMARY STEPS

1. **enable**
   - Enter your password if prompted.
2. **show policy-map** [*policy-map-name*]
   - (Optional) Enter the policy map name. The name can be a maximum of 40 alphanumeric characters.
3. **show policy-map interface**
4. **show running-config**
5. **exit**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br>• Enter your password if prompted.<br>**Example:**<br>`Router> enable` | Enables higher privilege levels, such as privileged EXEC mode. |
| Step 2 | **show policy-map** [*policy-map-name*]<br>• (Optional) Enter the policy map name. The name can be a maximum of 40 alphanumeric characters.<br>**Example:**<br>`Router# show policy-map unit-test` | (Optional) Displays the configuration of all classes for a specified policy map or of all classes for all existing policy maps. |
| Step 3 | **show policy-map interface**<br>**Example:**<br>`Router# show policy-map serial2/0` | (Optional) Displays the statistics and the configurations of the input and output policies that are attached to an interface. |
| Step 4 | **show running-config**<br>**Example:**<br>`Router# show running-config` | (Optional) Displays the contents of the currently running configuration file. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | **exit**<br>**Example:**<br>`Router(config-pmap-c)# exit` | Exits the policy-map class configuration mode. |

# Configuration Examples for Overhead Accounting

This example shows a two-level policy with Overhead Accounting on the parent shaper and a child that has bandwidth configured:

```
policy-map oh-child-bw
class oh-child
bandwidth percent 20

policy-map oh1
class class-default
shape average 1000000 account user-defined 33
service-policy oh-child-bw
```

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| Cisco IOS commands | Cisco IOS Master Commands List, All Releases |
| | |
| | |

### Standards and RFCs

| Standard/RFC | Title |
|---|---|
| | |
| | |

### MIBs

| MIB | MIBs Link |
|---|---|
| | To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL:<br>http://www.cisco.com/go/mibs |

### Technical Assistance

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

# Feature Information for Overhead Accounting

| Feature Name | Releases | Feature Information |
|---|---|---|
| Overhead Accounting | 15.2(1)T | Overhead accounting enables the router to account for packet overhead when shaping traffic to a specific rate. This accounting ensures that the router executes quality of service (QoS) features on the actual bandwidth used by subscriber traffic.<br>The following commands were introduced or modified: **shape** and **bandwidth**. |

CHAPTER 18

# Adaptive QoS over DMVPN

Adaptive QoS over Dynamic Multipoint VPN (DMVPN) ensures effective bandwidth management using dynamic shapers based on available bandwidth. This feature enables various QoS features to adapt to non service-level agreement (SLA) based environments where bandwidth is variable and fluctuate with time.

# 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 at the end of this module.

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

# Prerequisites for Adaptive QoS over DMVPN

Adaptive QoS over DMVPN can be enabled either on hub or spoke or both. To enable feature at a spoke side, the spoke must support basic egress per-SA QoS policy.

Internet Protocol Security (IPSec) is required and must be configured before Adaptive QoS is enabled on the DMVPN tunnel.

# Restrictions for Adaptive QoS over DMVPN

The Adaptive QoS over DMVPN feature configuration is:

- Supported only on DMVPN tunnels
- Allowed only on egress direction
- Allowed only in parent most policy that has class-default only

# Information About Adaptive QoS over DMVPN

## Overview of Adaptive QoS over DMVPN

Enterprise networks are increasingly using the Internet as form of WAN transport, therefore QoS models needs to be revisited. QoS works effectively when deployed in an service-level agreement (SLA) environment today, like Multiprotocol Label Switching (MPLS) . The available bandwidth on the internet at a given point of time can vary, and can be often much lesser than the actual bandwidth offered by the service provider. In cases of non SLA environments, QoS has limitations - mainly because it cannot predict changing bandwidth on the link.

Cisco Intelligent WAN (IWAN) recommends using Dynamic Multipoint VPN (DMVPN) over Internet to connect branches to the data center or headquarters, and QoS to be deployed in such environments of fluctuating bandwidth. Currently, the shapers that are applied as part of the egress QoS policy are static in value - they are configured based on the service provider bandwidth offering, they do not change with time and hence do not reflect the actual available Internet bandwidth. In many instances where Internet available bandwidth becomes much lesser than the offered bandwidth, the shapers become irrelevant as they do not adapt to the varying bandwidth. Due to the static value of the shapers, application traffic gets dropped indiscriminately at the Internet core, nullifying the very need to have configured a QoS policy to protect critical traffic.

DMVPN provides the ability to do QoS per-tunnel, which means a QoS policy can be applied at the hub towards a specific spoke, to ensure a high bandwidth hub does not overrun a low capacity spoke. However, these QoS policies still work with static shapers per spoke. If the bandwidth towards a particular spoke fluctuates, the shapers towards the spokes do not adapt. Also, it is not possible today to configure a QoS policy for the traffic from the spoke towards the hub, which is very common in many retail-like environments.

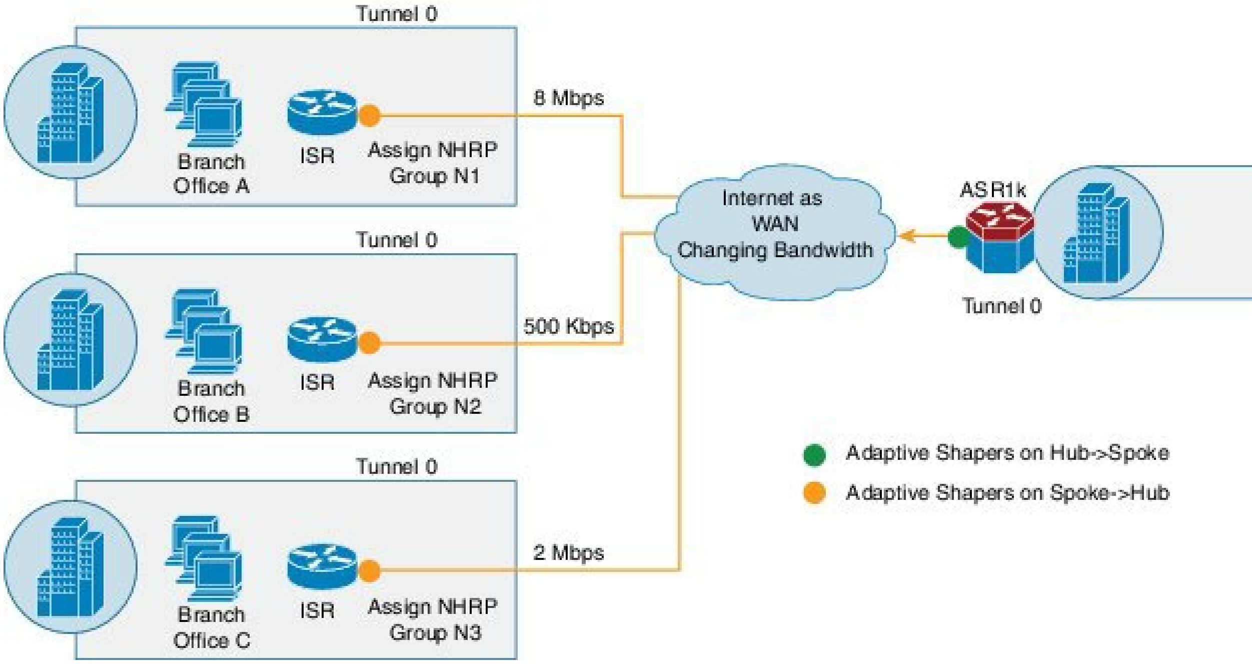
The Adaptive QoS over DMVPN feature provides the following benefits:

- Adjusts the shaper parameters based on the actual available Internet bandwidth in both directions that is periodically computed.
- Allows to configure a QoS policy on the spoke towards the hub.
- Ensures better control of application performance at the enterprise edge even in changing bandwidth scenarios over the Internet.
- Allows aggregate tunnel shape adaptation to provide effective bandwidth between spoke and hub.

# Adaptive QoS for Per-Tunnel QoS over DMVPN

Per-tunnel QoS over DMVPN can be configured on the hub towards the spoke today using Next Hop Resolution Protocol (NHRP) groups. The QoS policies contain static shapers. With Adaptive QoS, the framework of per tunnel QoS configuration remains the same, but the shaper can be an adaptive one as shown in the following figure. These shapers would adapt automatically based on the changing Internet bandwidth that is periodically computed using an algorithm.

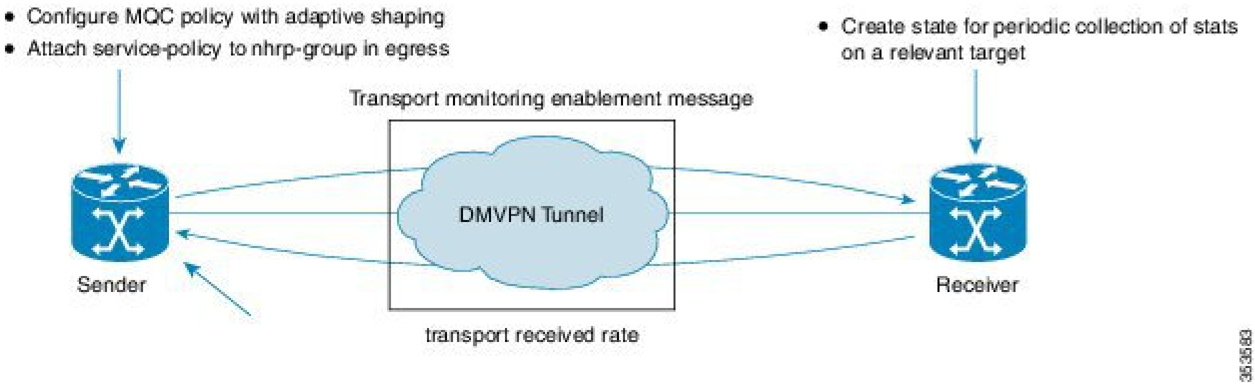
*Figure 5: Adaptive QoS for Per-Tunnel QoS over DMVPN*



### Workflow of Adaptive QoS

The Adaptive QoS over DMVPN feature adapts shaping rate at the Sender based on the available bandwidth between specific Sender and Receiver (two end-points of a DMVPN tunnel).

*Figure 6: Workflow of Adaptive QoS*


• Configure MQC policy with adaptive shaping
• Attach service-policy to nhrp-group in egress
• Create state for periodic collection of stats on a relevant target
Transport monitoring enablement message
DMVPN Tunnel
Sender
Receiver
transport received rate
353583

At the Sender:

- Configure MQC Policy with Adaptive shaping
- Attach service-policy to nhrp-group in Egress

At the Receiver:

Create state for periodic collection of stats on a relevant target

# How to Configure Adaptive QoS over DMVPN

**Note** Configure the Per-Tunnel QoS for DMVPN before configuring the Adaptive QoS over DMVPN feature, as Adaptive QoS over DMVPN feature is an enhancement to the Per-Tunnel QoS for DMVPN feature.

**Note** For details on configuring the Per-Tunnel QoS for DMVPN feature, refer to Per-Tunnel QoS for DMVPN
.

# Configuring Adaptive QoS for DMVPN

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *parent-policy-name*
4. **class class-default**
5. **shape adaptive** { **upper-bound** *bps* |**percent** *percentage* }[**lower-bound** *bps*| **percent** *percentage*]
6. **end**
7. **configure terminal**
8. **interface tunnel** *tunnel-id*
9. **nhrp map group** *group-name* **service-policy output** *parent-policy-name*
10. **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** *parent-policy-name*<br>**Example:**<br>`Router(config)# policy-map example` | Creates or modifies a child policy map and enters policy-map configuration mode.<br>• Enter the name of the child policy map. |
| **Step 4** | **class class-default**<br>**Example:**<br>`Router(config-pmap)# class class-default` | This step associates the traffic class with the traffic policy. Configures the default class map and enters policy-map class configuration mode. |
| **Step 5** | **shape adaptive** { **upper-bound** *bps* \|**percent** *percentage* }[**lower-bound** *bps*\| **percent** *percentage*] | Creates a specific adaptive shaper that has upper bound on the rate and optionally lower bound on the rate. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Router(config-pmap-c)# shape adaptive upper-bound 20000` | **Note** When such a template is attached to a target, adaptive shaping is enabled for that instance. Shaping rate adapts to a new rate, that is a function of parameters, including peer's received rate. |
| **Step 6** | **end**<br>**Example:**<br>`Router(config-pmap-c)# end` | Returns to privileged EXEC mode. |
| **Step 7** | **configure terminal**<br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 8** | **interface tunnel** *tunnel-id*<br>**Example:**<br>`Router(config)# interface tunnel 0` | Configures an interface type and enters interface configuration mode.<br>• Enter the interface type and interface number. |
| **Step 9** | **nhrp map group** *group-name* **service-policy output** *parent-policy-name*<br>**Example:**<br>`Router(config-if)# nhrp map group 1 service-policy output example` | Adds the NHRP group to the QoS policy map on the hub. |
| **Step 10** | **end**<br>**Example:**<br>`Router(config-if)# end` | Returns to privileged EXEC mode. |

# Verifying the Adaptive QoS over DMVPN

## SUMMARY STEPS

1. **enable**
2. **show dmvpn**
3. **show policy-map** [*policy-map-name*]
4. **show policy-map multipoint**
5. **exit**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br>**Example:**<br>`Router> enable` | Enables higher privilege levels, such as privileged EXEC mode.<br>• Enter your password if prompted. |
| Step 2 | **show dmvpn**<br>**Example:**<br>`Router# show dmvpn` | Displays detailed DMVPN information for each session, including the Next Hop Server (NHS) and NHS status, crypto session information, and socket details. Also displays the NHRP group received from the spoke and the QoS policy applied to the spoke tunnel. |
| Step 3 | **show policy-map** [*policy-map-name*]<br>**Example:**<br>`Router# show policy-map example` | Displays the configuration of all classes for a specified policy map or of all classes for all existing policy maps. |
| Step 4 | **show policy-map multipoint**<br>**Example:**<br>`Router# show policy-map tunnel 0` | (Optional) Displays the statistics and the configurations of the input and output policies that are attached to an interface. |
| Step 5 | **exit**<br>**Example:**<br>`Router(config-if)# exit` | (Optional) Returns to user EXEC mode. |

## Troubleshooting the Adaptive QoS over DMVPN

### SUMMARY STEPS

1. **enable**
2. **debug qos peer mon detail**
3. **debug qos peer rate detail**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Router> enable` | Enables higher privilege levels, such as privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **debug qos peer mon detail**<br>**Example:**<br>`Router# debug qos peer mon detail` | Displays debug messages for Adaptive QoS over DMVPN. |
| **Step 3** | **debug qos peer rate detail**<br>**Example:**<br>`Router# debug qos peer rate detail` | Displays debug messages for Adaptive QoS over DMVPN. |

# Configuration Examples for Configuring Adaptive QoS over DMVPN

## Example Configuring Adaptive QoS over DMVPN

The following example shows how to configure Adaptive QoS over DMVPN:

```
Router(config)# policy-map example
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape adaptive upper-bound 20000
Router(config-pmap-c)# end
Router# configure terminal
Router(config)# interface tunnel 0
Router(config-if)# nhrp map group 1 service-policy output example
Router(config-if)# end
```

# Example Verifying Adaptive QoS over DMVPN

The **show policy-map** and **show policy-map interface** commands can be used to confirm that the Adaptive QoS over DMVPN feature is enabled at an interface.

The following is a sample output of the **show dmvpn** command:

```
Router# show dmvpn


Interface: Tunnel1, IPv4 NHRP Details
Type: Hub, NHRP Peers:1,

# Ent  Peer NBMA Addr   Peer Tunnel Add   State UpDn Tm    Attrb
 ----- ------------- ----------------     ----- -------    -----
  1    10.1.1.1          10.10.1.2        UP    00:18:37   D


Interface: Tunnel2, IPv4 NHRP Details
Type: Hub, NHRP Peers:1,

# Ent Peer NBMA Addr   Peer Tunnel Add   State  UpDn Tm  Attrb
----- ---------------  --------------    ------  -------  -------

  1  10.2.1.1          10.10.2.2        UP     00:22:09  D


Interface: Tunnel3, IPv4 NHRP Details
Type: Hub, NHRP Peers:1,

# Ent Peer NBMA Addr   Peer Tunnel Add   State  UpDn Tm  Attrb
----- --------------   ---------------   ------ -------   ----
   1  10.3.1.1         10.10.3.2         UP     00:22:04   D


Interface: Tunnel4, IPv4 NHRP Details
Type: Hub, NHRP Peers:1,

# Ent Peer NBMA Addr   Peer Tunnel Add   State UpDn Tm  Attrb
----- --------------   ----------------  -----  ------  ----
  1    10.3.1.1        10.10.3.2         UP   00:22:01   D
```

The following is a sample output of the **show policy-map** command:

```
Router# show policy-map


Policy Map test
    Class class-default
      Adaptive Rate Traffic Shaping
      cir upper-bound 2120000 (bps) cir lower-bound 1120000 (bps)
```

The following is a sample output of the **show policy-map multipoint** command:

```
Router# show policy-map multipoint

 Service-policy output: test

  Class-map: class-default (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: any
  Queueing
  queue limit 64 packets
  (queue depth/total drops/no-buffer drops)0/0/0
  (pkts output/bytes output) 0/0
  shape (adaptive) cir 2120000,bc 8480, be 8480
  lower bound cir 2120000
  target shape rate 2120000
```

**Note** One of the important parameters displayed as an output of the **show policy-map multipoint** command is **target shape rate**. The Adaptive QoS over DMVPN feature dynamically changes the value of the **target shape rate** to adapt to the available bandwidth.

# Example for Troubleshooting Adaptive QoS over DMVPN

The **debug qos peer mon detail** and **debug qos peer rate detail** commands can be used to display any errors for the Adaptive QoS over DMVPN feature.

The following is a sample output of the **debug qos peer mon detail** command:

```
Router# debug qos peer mon detail

QoS peer remote monitoring debugging is on

Router#

*May 22 21:25:28.006 UTC: [SEND]Processing entry with address :
50.1.1.2,vrfid : 0 sending rate(delta bytes) : 1514
*May 22 21:25:28.006 UTC: [SEND]Processing entry with address :
50.1.1.3,vrfid : 0 sending rate(delta bytes) : 1598
*May 22 21:25:28.201 UTC: [RCV]Received message for interface Tunnel1
address 50.1.1.2 vrf 0
*May 22 21:25:28.201 UTC:
fdiff : 20517, sdiff : 19661, cur_dif : 3318, cum_diff : 20907

*May 22 21:25:28.201 UTC: qos_rate_status_update -- 392
*May 22 21:25:28.201 UTC: Last count : 128650
```

The following is a sample output of the **debug qos peer rate detail** command:

```
Router# debug qos peer rate detail


 *May 22 21:34:32.456 UTC: [RCV]Received message for interface Tunnel1
address 50.1.1.3 vrf 0
 *May 22 21:34:32.456 UTC: Enter qos_process_remote_rate_message:
 *May 22 21:34:32.456 UTC: Message for tun with o_ip : 50.1.1.3 tun t_ip
 : 13.1.1.1
 *May 22 21:34:32.456 UTC: [RCV]<DELTA>Message remote rate value is
116730f_cum_diff: 140155, s_cum_diff: 135612
 HoldTh: 5000, CurTh: 11250
 Gonna Go Up f_cum_diff: 140155, s_cum_diff: 135612
 Yes increasing
```

```
Suggested rate: 120000

*May 22 21:34:32.456 UTC: rx_bytes = 116730, tx_bytes = 125282, Suggested
rate = 120000
*May 22 21:34:32.456 UTC: Exiting : 1
```

# Additional References

The following sections provide references related to the Control Plane Logging feature.

### Related Documents

| Related Topic | Document Title |
|---|---|
| NHRP MIB | Dynamic Multipoint VPN Configuration Guide |
| QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | Cisco IOS Quality of Service Solutions Command Reference |
| QoS feature overview | Quality of Service Overview module |
| Per-Tunnel QoS for DMVPN | Dynamic Multipoint VPN Configuration Guide |

### Standards

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

### MIBs

| MIB | MIBs Link |
|---|---|
| CISCO-CLASS-BASED-QOS-MIB<br>CISCO-NHRP-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 |
|---|---|
| None | — |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content. | http://www.cisco.com/techsupport |

# Feature Information for Adaptive QoS over DMVPN

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 . An account on Cisco.com is not required.

*Table 10: Feature Information for Adaptive QoS over DMVPN*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Adaptive QoS over DMVPN | Cisco IOS 15.5(1)T | Adaptive QoS over Dynamic Multipoint VPN (DMVPN) ensures effective bandwidth management using dynamic shapers based on available bandwidth. This feature enables various QoS features to adapt to non service-level agreement (SLA) based environments where bandwidth is variable and fluctuate with time.<br>The following commands were introduced or modified: **shape adaptive**, **show policy-map**, and **show policy-map interface**. |