QoS: Policing and Shaping Configuration Guide, Cisco IOS XE 16 (Cisco NCS 4200 Series)

First Published: 2019-07-31

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

CONTENTS

CHAPTER 1

Feature History

The following table lists the new and modified features that are supported in the QoS: Policing and Shaping Configuration Guide in Cisco IOS XE 16 releases, on Cisco NCS 4201 and Cisco NCS 4202 routers.

Feature Name	Cisco IOS XE Release
Control Plane Policing Overview	16.11.1
Policer Adjustment in QoS Policy Map	16.11.1
QoS Overhead Accounting	16.11.1

The following table lists the new and modified features that are supported in the QoS: Policing and Shaping Configuration Guide in Cisco IOS XE 16 releases, on Cisco NCS 4206 and Cisco NCS 4216 routers.

Feature Name	Cisco IOS XE Release
Control Plane Policing Overview	16.11.1
Policer Adjustment in QoS Policy Map	16.11.1
QoS Overhead Accounting	16.11.1
CoPP	16.9.1

CHAPTER 2

Class-Based Policing

Class-based policing allows you to control the maximum rate of traffic that is 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.

Finding Feature Information

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

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

Information About Class-Based Policing

Class-Based Policing Functionality

The Class-Based Policing feature performs the following functions:

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

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 that contains the class-based policing configuration to an interface.

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.

Benefits of Class-Based Policing

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.

- 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.
- Use class-based policing to assign packets to a QoS group. The router uses the QoS group to determine how to prioritize packets.

Traffic can be marked without using the Class-Based Policing feature.

Restrictions for Class-Based Policing

- Class-based policing on sub-interfaces is *not* supported.
- Policing is supported for ingress policy maps only.
- Hierarchical policing (policing at both parent level and child level) is *not* supported. However, Egress two-level policer is supported provided PHB level priority policer is configured.
- Conditional marking is *not* supported.

How to Configure Class-Based Policing

Configuring a Traffic Policing Service Policy

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables higher privilege levels, such as privileged EXEC mode.

	Command or Action	Purpose
	`Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map** [**match-all** | **match-any**] *class-map-name* **Example:** `Router(config)# class-map match-any MATCH_PREC`	Specifies the name of the class map to be created and enters QoS class map configuration mode. • The class map defines the criteria to use to differentiate the traffic. For example, you can use the class map to differentiate voice traffic from data traffic, based on a series of match criteria defined using the **match** command. **Note** If the **match-all** or **match-any** keyword is not specified, traffic must match all the match criteria to be classified as part of the traffic class.
Step 4	**match ip precedence** *precedence-value* **Example:** `Router(config-cmap)# match ip precedence 0`	Enables packet matching on the basis of the IP precedence values you specify. **Note** You can enter up to four matching criteria, as number abbreviation (0 to 7) or criteria names (critical, flash, and so on), in a single match statement.
Step 5	**exit** **Example:** `Router(config-cmap)# exit`	Returns to global configuration mode.
Step 6	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map POLICE-SETTING`	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy, and enters QoS policy-map configuration mode.
Step 7	**class** {*class-name* | **class-default**} **Example:** `Router(config-pmap)# class MATCH_PREC`	Specifies the name of the class whose policy you want to create or change or specifies the default class (commonly known as the class-default class) before you configure its

	Command or Action	Purpose
		policy, and enters policy-map class configuration mode.
Step 8	**police** *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action* **violate-action** *action* **Example:** `Router(config-pmap-c)# police 8000 1000 1000 conform-action transmit exceed-action set-qos-transmit 1 violate-action drop`	Configures traffic policing according to burst sizes and any optional actions specified.
Step 9	**exit** **Example:** `Router(config-pmap-c)# exit`	(Optional) Exits policy-map class configuration mode.
Step 10	**exit** **Example:** `Router(config-pmap)# exit`	(Optional) Exits QoS policy-map configuration mode.
Step 11	**interface** *interface-type interface-number* **Example:** `Router(config)# interface GigabitEthernet 0/0/1`	Configures an interface type and enters interface configuration mode. • Enter the interface type and interface number.
Step 12	**service-policy** {**input** | **output**} *policy-map-name* **Example:** `Router(config-if)# service-policy input POLICE-SETTING`	Attaches a policy map to an interface. • Enter either the **input** or **output** keyword and the policy map name.
Step 13	**end** **Example:** `Router(config-if)# end`	(Optional) Exits interface configuration mode and returns to privileged EXEC mode.

Monitoring and Maintaining Traffic Policing

Procedure

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

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**show policy-map** **Example:** `Router# show policy-map`	Displays all configured policy maps.
Step 3	**show policy-map** *policy-map-name* **Example:** `Router# show policy-map pmap`	Displays the user-specified policy map.
Step 4	**show policy-map interface** **Example:** `Router# show policy-map interface`	Verifies that the Class-Based Policing feature is configured on your interface. If the feature is configured on your interface. • The command output displays policing statistics.

Verifying Class-Based Traffic Policing

Use the **show policy-map interface** 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.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show policy-map interface** **Example:** `Router# show policy-map interface`	Verifies that the Class-Based Policing feature is configured on your interface. If the feature is configured on your interface. • The command output displays policing statistics.
Step 3	**show policy-map interface** *type interface* **Example:** `Router# show policy-map interface GigabitEthernet 0/0/1`	Displays traffic statistics for policies applied to a specific interface.

	Command or Action	Purpose
Step 4	**show policy-map interface** *type interface* **service instance** *service-instance number* **Example:** `Router# show policy-map interface GigabitEthernet 0/0/1 service instance 1`	Displays the policy map information for a given service instance under an interface.
Step 5	**exit** **Example:** `Router# exit`	(Optional) Exits privileged EXEC mode.

Example: Verifying Class-Based Traffic Policing

```
Router# show policy-map interface
  FastEthernet1/1/1
  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 class-based policing is supported on your interface.

Configuration Examples for Class-Based Policing

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 the interface.

```
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 FastEthernet interface 1/1/1 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 is greater than the length of the packet (for example, B), then the packet conforms and B bytes should be removed from the bucket. 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 is less than the length of the packet, but the number of bytes in the exceed bucket is greater than the length of the packet (for example, B), the packet exceeds and B bytes are removed from the bucket.
- 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 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.

Verifying Class-Based Traffic Policing

Use the **show policy-map interface** 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:

```
Router# show policy-map interface
  FastEthernet1/1/1
  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
```

Use the **show policy-map interface** *type nummber* command to view the traffic statistics for policies applied to that specific interface:

```
Router# show policy-map interface gigabitethernet 0/0/1
 GigabitEthernet0/0/1

  Service-policy input: TUNNEL_MARKING

    Class-map: MATCH_PREC (match-any)
      72417 packets, 25418367 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 0
      QoS Set
        ip precedence tunnel 3
          Marker statistics: Disabled

    Class-map: MATCH_DSCP (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip dscp default (0)
      QoS Set
        ip dscp tunnel 3
          Marker statistics: Disabled

    Class-map: class-default (match-any)
      346462 packets, 28014400 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any

  Service-policy output: POLICE-SETTING

    Class-map: MATCH_PREC (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 0
      police:
          cir 8000 bps, bc 1000 bytes, be 1000 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          set-qos-transmit 1
        violated 0 packets, 0 bytes; actions:
```

```
          drop
        conformed 0000 bps, exceed 0000 bps, violate 0000 bps

    Class-map: class-default (match-any)
      31 packets, 2019 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

Use the **show policy-map interface service instance** command to view the traffic statistics for policy applied to the specific service instance in that specific interface:

```
Router# show policy-map interface gig0/0/1 service instance 10
 GigabitEthernet0/0/1: EFP 10

       Service-policy input: ac1

 Class-map: ac1 (match-all)
   0 packets, 0 bytes
   5 minute offered rate 0000 bps, drop rate 0000 bps
   Match: access-group 1
   police:
       cir 50000000 bps, bc 1562500 bytes
     conformed 0 packets, 0 bytes; actions:
       transmit
     exceeded 0 packets, 0 bytes; actions:
       drop
     conformed 0000 bps, exceeded 0000 bps

 Class-map: class-default (match-any)
   0 packets, 0 bytes
   5 minute offered rate 0000 bps, drop rate 0000 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*
Traffic marking	“Marking Network Traffic” module
Traffic policing	“Traffic Policing” module
Traffic policing and shaping concepts and overview information	“Policing and Shaping Overview”
Modular Quality of Service Command-Line Interface (MQC)	“Applying QoS Features Using the MQC” module

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
Class-Based Quality of Service MIB • CISCO-CLASS-BASED-QOS-MIB • CISCO-CLASS-BASED-QOS-CAPABILITY-MIB	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: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 2697	*A Single 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 3

Port-Shaper and LLQ in the Presence of EFPs

The Port-Shaper and LLQ in the Presence of EFPs feature allows network designers to configure port and class policies on ports that contain Ethernet Flow Points (EFPs). These policies support Low Latency Queueing (LLQ) and traffic prioritization across the EFPs.

Finding Feature Information

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

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

Restrictions for Port-Shaper and LLQ in the Presence of EFPs

- If you configure port level shaper with the policy applied at EFP level then port shaper does not work. However, 3 level HQoS policy with port and logical shaper can be applied at the EFP level. Logical shaper configured at logical level does work but port shaper does not work.
- If you configure a class-based HQOS or LLQ policy on the port, you cannot configure service-policies on Ethernet Flow Points (EFPs). The only exception to this is the class-default shaper policy and match EFP policy.
- If you configure a class-based policy on the port, you cannot configure service-policies on EFPs.
- If you configure a class-default port-shaper based policy on the port, you can configure service-policy on EFPs.

- Usage of bandwidth remaining percentage (BRP) in the absence of priority class, allocates the available bandwidth in an iterative way. For example, the bandwidth is allocated for the first BRP class as per the percentage of share configured in the respective class-map and the remaining bandwidth is iteratively allocated to all other BRP classes until the bandwidth is exhausted.

Information About Port-Shaper and LLQ in the Presence of EFPs

Ethernet Flow Points and LLQ

An Ethernet Flow Point (EFP) is a forwarding decision point in the provider edge (PE) router, which gives network designers flexibility to make many Layer 2 flow decisions within the interface. Many EFPs can be configured on a single physical port. (The number varies from one device to another.) EFPs are the logical demarcation points of an Ethernet virtual connection (EVC) on an interface. An EVC that uses two or more User-Network Interfaces (UNIs) requires an EFP on the associated ingress and egress interfaces of every device that the EVC passes through.

The Egress HQoS with Port Level Shaping feature allows network designers to configure port and class policies on ports that contain EFPs. These policies support Low Latency Queueing (LLQ) and traffic prioritization across the EFPs.

For information on how to configure LLQ, see the *QoS Congestion Management Configuration Guide*.

How to Configure Port-Shaper and LLQ in the Presence of EFPs

To configure the Port-Shaper and LLQ in the Presence of EFPs feature, you first create either a hierarchical or flat policy map that supports Low Latency Queueing (LLQ), which you then attach to an EFP interface.

Configuring Hierarchical Policy Maps

To configure hierarchical policy maps, you create child policies which you then attach to a parent policy. The parent policy is then attached to an interface.

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 **policy-map** *policy-map-name*

Example:

```
Device(config)# policy-map child-llq
```

Creates or modifies the child policy and enters QoS policy-map configuration mode.

- child-llq is the name of the child policy map.

Step 4 **class** *class-map-name*

Example:

```
Device(config-pmap)# class precedenc-1
```

Assigns the traffic class you specify to the policy map and enters QoS policy-map class configuration mode.

- precedenc-1 is the name of a previously configured class map and is the traffic class for which you want to define QoS actions.

Step 5 **set cos** *value*

Example:

```
Device(config-pmap-c)# set cos 5
```

(Optional) Sets the Layer 2 class of service (CoS) value of an outgoing packet.

- The value is a specific IEEE 802.1Q CoS value from 0 to 7.

Step 6 **bandwidth percent** *percent*

Example:

```
Device(config-pmap-c)# bandwidth percent 20
```

(Optional) Specifies a bandwidth percent for class-level queues to be used during congestion to determine the amount of excess bandwidth (unused by priority traffic) to allocate to nonpriority queues.

Step 7 **exit**

Example:

```
Device(config-pmap-c)# exit
```

Exits QoS policy-map class configuration mode.

Step 8 **class** *class-map-name*

Example:

```
Device(config-pmap)# class precedenc-2
```

Assigns the traffic class you specify to the policy map and enters QoS policy-map class configuration mode.

- precedenc-2 is the name of a previously configured class map and is the traffic class for which you want to define QoS actions.

Step 9 **bandwidth percent** *percent*

Example:

```
Device(config-pmap-c)# bandwidth percent 80
```

(Optional) Specifies a bandwidth percent for class-level queues to be used during congestion to determine the amount of excess bandwidth (unused by priority traffic) to allocate to nonpriority queues.

Step 10 **exit**

Example:

```
Device(config-pmap-c)# exit
```

Exits QoS policy-map class configuration mode.

Step 11 **policy-map** *policy-map-name*

Example:

```
Device(config-pmap)# policy-map parent-llq
```

Creates or modifies the parent policy.

- parent-llq is the name of the parent policy map.

Step 12 **class** *class-default*

Example:

```
Device(config-pmap)# class class-default
```

Configures or modifies the parent class-default class and enters QoS policy-map class configuration mode.

- You can configure only the class-default class in a parent policy. Do not configure any other traffic class.

Step 13 **service-policy** *policy-map-name*

Example:

```
Device(config-pmap-c)# service-policy child-llq
```

Applies the child policy to the parent class-default class.

- child-llq is the name of the child policy map configured in step 1.

Configuring Class-default Port-Shaper Policy Maps

To configure hierarchical policy maps, first create the child policies and then attach it to a parent policy. The parent policy must be attached to an interface.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Device(config)# policy-map child-llq`	Creates or modifies the child policy and enters QoS policy-map configuration mode. • child-llq is the name of the child policy map.
Step 4	**class** *class-default* **Example:** `Device(config-pmap)# class class-default`	Configures or modifies the parent class-default class and enters QoS policy-map class configuration mode. • You can configure only the class-default class in a parent policy. Do not configure any other traffic class.
Step 5	**shape-average** *shape-value* **Example:** `Device(config-pmap-c)#shape average 200000000`	Configures a shape entity with a Comitted Information Rate of 200 Mb/s.
Step 6	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits QoS policy-map class configuration mode.

Configuring Port-Shaper Policy Maps

Procedure

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

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Device(config)# policy-map def`	Creates or modifies the child policy and enters QoS policy-map configuration mode.
Step 4	**class** *class-default* **Example:** `Device(config-pmap)# class class-default`	Assigns the traffic class you specify to the policy map and enters QoS policy-map class configuration mode.
Step 5	**shape-average** *shape-value* **Example:** `Device(config-pmap-c)#shape average 200000000`	Configures a shape entity with a Comitted Information Rate of 200 Mb/s.
Step 6	**service-policy** *policy-map-name* **Example:** `Device(config-pmap-c)# service-policy child-llq`	Applies the child policy to the parent class-default class. • child-llq is the name of the child policy map configured in Configuring Class-default Port-Shaper Policy Maps, on page 16.

Configuring an LLQ Policy Map

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 **policy-map** *policy-map-name*

Example:

```
Device(config)# policy-map llq-flat
```

Creates a policy and enters QoS policy-map configuration mode.

Step 4 **class** *class-map-name*

Example:

Assigns the traffic class you specify to the policy map and enters policy-map class configuration mode.

Step 5 **priority**

Example:

```
Device(config-pmap-c)# priority
```

Configures LLQ, providing strict priority queueing (PQ) for class-based weighted fair queueing (CBWFQ).

Step 6 **exit**

Example:

```
Device(config-pmap-c)# exit
```

Exits QoS policy-map class configuration mode.

Step 7 **class** *class-map-name*

Example:

Assigns the traffic class you specify to the policy map and enters QoS policy-map class configuration mode.

Step 8 **shape average** *value*

Example:

```
Device(config-pmap-c)# shape average 200000000
```

Configures a shape entity with a Comitted Information Rate of 200 Mb/s.

Step 9 **exit**

Example:

```
Device(config-pmap-c)# exit
```

Exits QoS policy-map class configuration mode.

Step 10 **class** *class-map-name*

Example:

Assigns the traffic class you specify to the policy map and enters QoS policy-map class configuration mode.

Step 11 **bandwidth** *percent*

Example:

```
Device(config-pmap-c)# bandwidth 4000000
```

(Optional) Specifies a bandwidth percent for class-level queues to be used during congestion to determine the amount of excess bandwidth (unused by priority traffic) to allocate to non-priority queues.

Step 12 **exit**

Example:

```
Device(config-pmap-c)# exit
```

Exits QoS policy-map class configuration mode.

Configuring Port Level Shaping on the Main Interface with Ethernet Flow Points

To configure port level shaping on the main interface with EFPS, first you enable the autonegotiation protocol on the interface, then you attach a policy map to the interface and finally you configure the Ethernet service instance.

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 **interface** *type number*

Example:

```
Device(config)# interface GigabitEthernet 0/0/1
```

Configures an interface type and enters interface configuration mode.

- Enter the interface type number.

Step 4 **no ip address**

Example:

```
Device(config-if)# no ip address
```

Disables IP routing on the interface.

Step 5 **negotiation auto**

Example:

```
Device(config-if)# negotiation auto
```

Enables the autonegotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.

Step 6 **service-policy output** *policy-map-name*

Example:

```
Device(config-if)# service-policy output parent-llq
```

Specifies the name of the policy map to be attached to the input or output direction of the interface.

- You can enter the name of a hierarchical or a flat policy map.

Step 7 **service instance** *id* **ethernet**

Example:

```
Device(config-if)# service instance 1 ethernet
```

Configures an Ethernet service instance on an interface and enters service instance configuration mode.

Step 8 **encapsulation dot1q** *vlan-id*

Example:

```
Device(config-if-srv)# encapsulation dot1q 100
```

Defines the matching criteria to map 802.1Q frames' ingress on an interface to the service instance.

Step 9 **bridge-domain** *bridge-domain-id*

Example:

```
Device(config-if-srv)# bridge-domain 100
```

Binds the bridge domain to the service instance.

Step 10 **exit**

Example:

```
Device(config-if-serv)# exit
```

Exits service instance configuration mode.

Step 11 **service instance** *id* **ethernet**

Example:

```
Device(config-if)# service instance 2 ethernet
```

Configures an Ethernet service instance on an interface and enters service instance configuration mode.

Step 12 **encapsulation dot1q** *vlan-id*

Example:

```
Device(config-if-srv)# encapsulation dot1q 101
```

Defines the matching criteria to map 802.1Q frames' ingress on an interface to the service instance.

Step 13 **bridge-domain** *bridge-domain-id*

Example:

```
Device(config-if-srv)# bridge-domain 101
```

Binds the bridge domain to the service instance.

Step 14 **exit**

Example:

```
Device(config-if-srv)# exit
```

Exits QoS policy-map class configuration mode.

Step 15 **end**

Example:

```
Device(config-if)# end
```

(Optional) Exits interface configuration mode.

Configuration Examples for Port-Shaper and LLQ in the Presence of EFPs

Example: Configuring Hierarchical QoS Port Level Shaping on the Main Interface with EFPs

The following example shows how to configure hierarchical QoS port level shaping on a main physical interface to support traffic prioritization and Low Level Queueing across all EFPs configured on the interface:

```
policy-map parent-llq
 class class-default
   service-policy child-llq
```

```
policy-map child-llq
 class precedenc-1
  set cos 5
  bandwidth percent 20
 class precedenc-2
  bandwidth percent 80

interface GigabitEthernet 0/0/1
 no ip address
 negotiation auto
 service-policy output parent-llq
 service instance 1 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 2 ethernet
  encapsulation dot1q 101
  bridge-domain 101
```

Note Only match EFP and match qos-group is supported on RSP3 in egress policy map.

Configuration Example: Class-default Port-Shaper and EFP policy

The following example shows how to configure class-default port-shaper and EFP policy, where the main interface can have the class-default shaper policy and EFP can have the HQOS policies.

```
policy-map co12
class class-default
shape average 50m


policy-map def
 class class-default
shape average 500m
service-policy co12
```

Example: Configuring Port Level Shaping on the Main Interface with EFPs

The following example shows how to configure port level shaping on a main physical interface to support traffic prioritization and Low Level Queueing across all Ethernet Flow Points (EFPs) configured on the interface:

```
policy-map llq_flat
 class dscp-af1
  priority
  class dscp-af2
   shape average 200000000
  class dscp-af3
   bandwidth 400000

interface GigabitEthernet 0/0/1
 no ip address
```

```
negotiation auto
service-policy output llq_flat
service instance 1 ethernet
 encapsulation dot1q 100
 bridge-domain 100
!
service instance 2 ethernet
 encapsulation dot1q 101
 bridge-domain 101
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS QoS Command Reference*
Policing and shaping	"Policing and Shaping Overview" module
Class maps	"Applying QoS Features Using the MQC" module
Policy maps	"Applying QoS Features Using the MQC" module
Low Latency Queueing	*QoS Congestion Management Configuration Guide*

Standards and RFCs

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

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified.	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: http://www.cisco.com/go/mibs

Technical Assistance

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

CHAPTER 4

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 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.

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 Control Plane Policing

Input Rate-Limiting Support

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

MQC Restrictions

The Control Plane Policing feature requires the Modular QoS CLI (MQC) to configure packet classification and traffic policing. All restrictions that apply when you use the MQC to configure traffic policing also apply when you configure control plane policing.

Match Criteria Support

Only the extended IP access control lists (ACLs) classification (match) criteria is supported.

Restrictions for CoPP on the RSP3

- sdm prefer enable_copp template must be enabled on the the RSP3 module to activate COPP.
- Ingress and Egress marking are not supported.
- Egress COPP is not supported. COPP with marking is not supported.
- CPU bound traffic (punted traffic) flows is supported via the same queue with or without CoPP.
- Only match on access group is supported on a CoPP policy.
- Hierarchical policy is not supported with CoPP.
- Class-default is not supported on CoPP policy.
- User defined ACLs are not subjected to CoPP classified traffic.
- A CoPP policy map applied on a physical interface is functional.
- When COPP template is enabled, classification on outer Vlan, inner Vlan, Inner Vlan Cos, destination MAC address, source IP address, and destination IP address are not supported.

  The template-based model is used to enable COPP features and disable some of the above mentioned QOS classifications.
- When sdm prefer enable_copp template is enabled, sdm prefer enable_match_inner_dscp template is not supported.
- Only IP ACLs based class-maps are supported. MAC ACLs are not supported.
- Multicast protocols like PIM, IGMP are not supported.
- Only CPU destined Unicast Layer3 protocols packets are matched as part of COPP classification.

Restrictions on Firmware

- Port ranges are not supported.
- Only exact matches are supported, greater than, less than and not equal are not supported.
- Internet Control Message Protocol (ICMP) inner type’s classification not supported.
- Match any is only supported at class-map level.
- Policing action is supported on a CoPP policy map.

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

Control Plane Terms to Understand

On the router, the following terms are used for the Control Plane Policing feature:

- Control plane—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 XE functions. The traffic sent to or sent by the control plane is called control traffic.
- Forwarding plane—A device that is responsible for high-speed forwarding of IP packets. Its logic is kept simple so that it can be implemented by hardware to do fast packet-forwarding. It punts packets that require complex processing (for example, packets with IP options) to the RP for the control plane to process them.

Control Plane Policing Overview

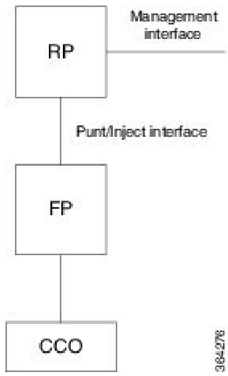
To protect the control plane on a router from DoS attacks and to provide fine-control over the traffic to the control plane, the Control Plane Policing feature treats the control plane as a separate entity with its own interface for ingress (input) and egress (output) traffic. This interface is called the punt or inject interface, and it is similar to a physical interface on the router. Along this interface, packets are punted from the forwarding plane to the RP (in the input direction) and injected from the RP to the forwarding plane (in the output direction). A set of quality of service (QoS) rules can be applied on this interface (in the input direction) in order to achieve CoPP.

These QoS rules are applied only after the packet has been determined to have the control plane as its destination. You can configure a service policy (QoS policy map) to prevent unwanted packets from progressing after a specified rate limit has been reached; for example, a system administrator can limit all TCP/TELNET packets that are destined for the control plane.

You can use the **platform qos-feature copp-mpls enable** command to enable the Control Plane Policing feature on the device for MPLS explicit null scenario, control packets destined to the device is punted to proper control CPU Q. If CoPP-MPLS remains disabled, then self destined control packets like BGP, LDP, telnet and so on, that are MPLS explicit null tagged are not classified by CoPP and is punted to HOST_Q instead of CFM_Q/CONTROL_Q.

> **Note**
> The command **platform qos-feature copp-mpls enable** is supported only on Cisco NCS 4200 platform.

Figure 1: Abstract Illustration of a Router with a Single RP and Forwarding Plane



The figure provides an abstract illustration of the router with a single RP and forwarding plane. Packets that are destined to the control plane come in through the carrier card and then go through the forwarding plane before being punted to the RP. When an input QoS policy map is configured on the control plane, the forwarding plane performs the QoS action (for example, a transmit or drop action) before punting packets to the RP in order to achieve the best protection of the control plane in the RP.

> **Note**
> As mentioned in this section, the control plane interface is directly connected to the RP, so all traffic through the control plane interface to or from the control-plane is not subject to the CoPP function performed by the forwarding plane.

Supported Protocols

The following table lists the protocols supported on Control Plane Policing feature.

Supported Protocols	**Criteria**	**Match**	**Queue#**
TFTP - Trivial FTP	Port Match	IP access list ext copp-system-acl-tftp permit udp any any eq 69	NQ_CPU_HOST_Q
TELNET	Port Match	IP access list ext copp-system-acl-telnet permit tcp any any eq telnet	NQ_CPU_CONTROL_Q
NTP - Network Time Protocol	Port Match	IP access list ext copp-system-acl-ntp permit udp any any eq ntp	NQ_CPU_HOST_Q

Supported Protocols	Criteria	Match	Queue#
FTP - File Transfer Protocol	Port Match	IP access list ext copp-system-acl-ftp permit tcp host any any eq ftp	NQ_CPU_HOST_Q
SNMP - Simple Network Management Protocol	Port Match	IP access list ext copp-system-acl-snmp permit udp any any eq snmp	NQ_CPU_HOST_Q
TACACS - Terminal Access Controller Access-Control System	Port Match	IP access list ext copp-system-acl-tacacs permit tcp any any tacacs	NQ_CPU_HOST_Q
FTP-DATA	Port Match	IP access list ext copp-system-acl-ftpdata permit tcp any any eq 20	NQ_CPU_HOST_Q
HTTP - Hypertext Transfer Protocol	Port Match	IP access list ext copp-system-acl-http permit tcp any any eq www	NQ_CPU_HOST_Q
WCCP - Web Cache Communication Protocol	Port Match	IP access list ext copp-system-acl-wccp permit udp any eq 2048 any eq 2048	NQ_CPU_HOST_Q
SSH - Secure Shell	Port Match	IP access list ext copp-system-acl-ssh permit tcp any any eq 22	NQ_CPU_HOST_Q
ICMP - Internet Control Message Protocol	Protocol Match	IP access list copp-system-acl-icmp permit icmp any any	NQ_CPU_HOST_Q
DHCP - Dynamic Host Configuration Protocol	Port Match	IP access list copp-system-acl-dhcp permit udp any any eq bootps	NQ_CPU_HOST_Q
MPLS- OAM	Port Match	IP access list copp-system-acl-mplsoam permit udp any eq 3503 any	NQ_CPU_HOST_Q

Supported Protocols	Criteria	Match	Queue#
LDP - Label Distribution Protocol	Port Match	IP access list copp-system-acl-ldp permit udp any eq 646 any eq 646 permit tcp any any eq 646	NQ_CPU_CFM_Q
RADIUS - Remote Authentication Dial In User Service	Port Match	IP access list copp-system-radius permit udp any any eq 1812 permit udp any any eq 1813 permit udp any any eq 1645 permit udp any any eq 1646 permit udp any eq 1812 any permit udp any eq 1813 any permit udp any eq 1645 any	NQ_CPU_HOST_Q

Input Rate-Limiting and Silent Mode Operation

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

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

How to Use Control Plane Policing

Defining Control Plane Services

Perform this task to define control plane services, such as packet rate control and silent packet discard for the RP.

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.

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control plane interface.
- Input policing does not provide any performance benefits. It simply controls the information that is entering the device.

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 **control-plane**

Example:

```
Device(config)# control-plane
```

Enters control-plane configuration mode (which is a prerequisite for defining control plane services).

Step 4 **service-policy** [**input** |**output**] *policy-map-name*

Example:

```
Device(config-cp)# service-policy input control-plane-policy
```

Attaches a QoS service policy to the control plane.

- **input**—Applies the specified service policy to packets received on the control plane.
- *policy-map-name*—Name of a service policy map (created using the **policy-map** command) to be attached.

Step 5 **end**

Example:

```
Device(config-cp)# end
```

(Optional) Returns to privileged EXEC mode.

Verifying Control Plane Services

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **show policy-map control-plane** [**all**] [**input** |**output** [**class** *class-name*]]

Example:

```
Device# show policy-map control-plane all
```

Displays information about the control plane.

- **all**—(Optional) Displays service policy information about all QoS policies used on the CP.
- **input**—(Optional) Displays statistics for the attached input policy.
- **class** *class-name*—(Optional) Specifies the name of the traffic class whose configuration and statistics are displayed.

Step 3 **exit**

Example:

```
Device# exit
```

(Optional) Exits privileged EXEC mode.

Examples

The following example shows that the policy map TEST is associated with the control plane.

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

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

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

Step 2 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 **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*

Example:

Configures an access list for filtering frames by UDP protocol and matches only packets with a given port number.

Step 4 **class-map [match-any | match-all | type]** *class-map-name*

Example:

```
Device(config)# class-map match-any MyClassMap
```

Creates a class-map and enters QoS class-map configuration mode.

Step 5 **match access-group** [*access-list-index* | *access-group-name*]

Example:

```
Device(config-cmap)# match access-group 111
```

Specifies access groups to apply to an identity policy. The range of valid values is 1-2799.

Step 6 **exit**

Example:

```
Device(config-cmap)# exit
```

Exits QoS class-map configuration mode and returns to global configuration mode.

Step 7 **policy-map** *policy-map-name*

Example:

```
Device(config)# policy-map Policy1
```

Specifies a service policy and enters QoS policy-map configuration mode.

Step 8 **class** [*class-map-name* | *class-default*]

Example:

```
Device(config-pmap)# class MyClassMap
```

Enters QoS policy-map class configuration more

Step 9 **police** {*rate-bps* | **cir** {*cir-bps* | **percent** *percent*}} [**bc** *burst-bytes*] [**conform-action** | **exceed-action** | **violate-action**]*action*] [ ]

Example:

Configure a traffic policer based on the traffic rate or committed information rate (CIR). By default, no policer is defined.

- *rate-bps*—Specifies average traffic rate in bits per second (b/s). The range is 64000 to 10000000000. Supply an optional postfix (K, M, G). Decimal point is allowed.
- **cir**—Specifies a committed information rate (CIR).
- *cir-bps*—Specifies a CIR in bits per second (b/s). The range is 64000 to 10000000000. Supply an optional postfix (K, M, G). Decimal point is allowed.
- **be** *burst-bytes*—(Optional) Specifies the conformed burst (be) or the number of acceptable burst bytes. The range is 8000 to 16000000.
- **conform-action** *action*— (Optional) Specifies action to take on packets that conform to the specified rate limit.
- **pir** *pir-bps*—(Optional) Specifies the peak information rate (PIR).

Note **cir percent** *percent* option is not supported on the router.

Step 10 **exit**

Example:

```
Device(config-pmap-c-police)# exit
```

Exits policy-map class police configuration mode

Step 11 **exit**

Example:

```
Device(config-pmap-c)# exit
```

Exits policy-map class configuration mode

Step 12 **exit**

Example:

```
Device(config-pmap)# exit
```

Exits policy-map configuration mode

Step 13 **control-plane**

Example:

```
Device(config)# control-plane
```

Enters control plane configuration mode.

Step 14 **service-policyinput** *policy-map-name*

Example:

```
Device(config-cp)# service-policy input Policy1
```

Attaches a policy map to a control plane.

Step 15 **exit**

Example:

```
Device(config-cp)# exit
```

Exits control plane configuration mode and returns to global configuration mode.

Step 16 **exit**

Example:

```
Device(config)# exit
```

Exits global configuration mode returns to privileged EXEC mode.

Configuration Examples for Control Plane Policing

Example: Configuring Control Plane Policing on Input Telnet Traffic

Verification Examples for CoPP

The following example shows how to verify control plane policing on a policy map.

```
Router# show policy-map control-plane
           Control Plane
         Service-policy input: control-plane-in
         Class-map: telnet-class (match-all)
           10521 packets, 673344 bytes
           5 minute offered rate 18000 bps, drop rate 15000 bps
           Match: access-group 102
           police:  cir 64000 bps, bc 8000 bytes
           conformed 1430 packets, 91520 bytes; actions:
           transmit
           exceeded 9091 packets, 581824 bytes; actions:
           drop
          conformed 2000 bps, exceeded 15000 bps
     Class-map: class-default (match-any)
           0 packets, 0 bytes
           5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: any
```

The following command is used to verify the TCAM usage on the router.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mcl/allreleasemcl/all-book.html

Standards and RFCs

Standard/RFC	Title
No specific Standards and RFCs are supported by the features in this document.	—

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: http://www.cisco.com/go/mibs

Technical Assistance

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

CHAPTER 5

QoS 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 that is used by subscriber traffic.

The overhead accounting feature enables the router to account for downstream Ethernet frame headers when applying shaping to packets. The traffic scheduler allows a minimum amount of value more than the configured rate at the port, in addition to the excess bytes configured on that port.

Restrictions for QoS Overhead Accounting

- Accounting feature is supported only for the following scenarios:
  - Layer 2 interface
  - MPLS imposition
  - MPLS disposition
- Accounting feature can be enabled per interface and only one value of compensation bytes can be configured globally.
- The feature is applied in the following scenarios:
  - Per interface QoS overhead accounting can take effect only during a policy-map detach or attach process.
  - Any dynamic modification, for example, enabling or disabling on an interface or change in global compensation bytes can reflect per interface only after a policy-map detach or attach process.
  - Already configured policy-map on the accounting enabled interface needs to be detached and reattached.

- While detaching, ensure to perform the following tasks:
    - Detach the policy-map per interface.
    - Disable the accounting feature for that interface.
    - Re-attach the policy-map based on the requirement.
- QoS overhead accounting is not supported for port channel interface and member links.
- QoS overhead accounting is not supported for trunk EFPs on an interface.
- Accounting is not supported if interface has Ethernet loopback that is enabled.

How to Configure QoS Overhead Accounting

Applying Overhead Accounting on a Particular Interface

To apply overhead accounting on a particular interface, for example layer 2 interface and MPLS disposition, enter the following commands:

```
Router> enable
Router# configure terminal
Router(config)# qos-overhead-accounting enable gi 0/0/0
```

Configuring Number of Bytes to be Accounted

To configure the number of bytes that need to be accounted, enter the following commands:

```
Router> enable
Router# configure terminal
Router(config)# qos-overhead-accounting positive 8
```

Configuring Overhead Accounting for MPLS Imposition

To configure compensation for the MPSL imposition with access interface as gig 0/0/0 and core port as gig 0/0/1, enter the following steps:

```
Router> enable
Router# configure terminal
Router(config)# qos-overhead-accounting enable gi 0/0/1
Router(config)# qos-overhead-accounting positive 8
Router(config)# qos-overhead-accounting enable gi 0/0/0
```

To disable the compensation, enter the following commands:

```
Router> enable
```

```
Router# configure terminal
Router(config)#no qos-overhead-accounting enable gi 0/0/1
Router(config)#no qos-overhead-accounting enable gi 0/0/0
```

Verifying Overhead Accounting compensation

Use the following show command to display the set of interfaces on which overhead accounting is enabled:

```
Router#show platform hardware pp active feature qos oh-accounting interface all
Overhead Accounting Target Info
Interface Name                GID    Status    Bytes (Shadow)   Bytes (Actual)
---------------------------------------------------------------------------------
GigabitEthernet0/0/0          269    Enabled    8                 8
```

CHAPTER 6

Policer Adjustment in QoS Policy Map

Policers are configured usually at a value range of 64,000–10 G whereas the hardware policer is programmed only to discrete value. The policer rate received is less than that of the configured CIR and PIR values. The policer adjustment feature is added to adjust the CIR and PIR values of hardware policer either to match the configured value or to the next higher value available in hardware.

The policer adjustment feature is supported on the RSP2 module.

To enable policer adjustment, use the **platform qos-adjust-policer enable** at the global configuration mode for a table map. You can view the **show platform hardware pp active feature QoS interface** command to compare the configured values of CIR and PIR values in the qos-policy and the actual programmed values in hardware.

With the policer adjustment feature, the policer rate is compensated with + 0 to + 0.5 to the configured policer rate so that you can achieve the received rate more than or equal to that of the configured rate.

Restrictions for Policer Adjustment

- Policy adjustment is performed at a global configuration level and it is not supported on each port or EFP.
- Detaching and attaching of policer from ports after applying the policy adjustment feature at a global configuration works for applied ports. For the remaining ports to which detaching and attaching is not performed after enabling the policy adjustment works in a legacy QoS functionality manner.
- Policer enhancement is supported on EFP, TEFP, routed port, and port channel.
- BC or BE values are not adjusted, and only CIR and PIR or EIR are adjusted. Even if BC or BE values are configured, the values that are displayed in the show command do not match exactly with IOS values.
- CIR rates 64,000–3,00,000 can have rates more than 0.5 percent as this rate limits to already available percent and effects higher rates.

How to configure Policer Adjustment

Enabling Policer Adjustment

To enable a policer adjustment at the global configuration mode, enter the following command:

```
Router> enable
Router# configure terminal
Router(config)# platform qos-adjust-policer enable
```

After enabling the policer adjustment, you must detach and attach the policer from port, then only the feature is applied on the port.

Disabling Policer Adjustment

To disable the policer adjustment globally, enter the no form of the following command:

```
{no} platform qos-adjust-policer enable
```

After disabling the policer adjustment, you need to detach and attach the existing policy-map from the port or service and then only the policer adjustment is disabled.

Verifying Policer Adjustment

Use the following **show platform hardware pp active feature QoS interface {intf_name} {service-instance} {EVC_num} input/ouput** command to view the configured and programmed policer values:

```
Router# show platform hardware pp active feature qos interface te 0/0/13 ser 2 in

Policy details:

Interface: TenGigabitEthernet0/0/13
Policy: TMO-EVC
Service instance number: 2
Direction: input
-------------------------------------------------------------------
Class: EVC, Level: 2
Policer Mode: IETF_2R3C
Policer Index Id: 33
Policer Profile Id: 12
Policer feature         Software value            Asic value
CIR                        5000000 kbps             5062500 kbps
PIR                        7000000 kbps                  NA
EIR (PIR - CIR)            2000000 kbps             2024884 kbps
BC                         2500000 bytes            2500000 bytes
BE                        16000000 bytes           16000000 bytes
```