# Cisco Nexus 1000V for Microsoft Hyper-V QoS Configuration Guide, Release 5.x

**First Published:** December 15, 2014

**Last Modified:** October 02, 2015

**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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: http://www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2015 Cisco Systems, Inc. All rights reserved.

# CONTENTS


CHAPTER 1 **Overview** 1

Information About Quality of Service 1

Traffic Classification and Marking 2

QoS Commands 2

Default QoS Behavior 3

Supported RFCs 3

High Availability Requirements for QoS Features 4

Commonly Used DSCP Values 4

IP Precedence Values 5

CHAPTER 2 **Configuring QoS Classification** 7

Information About Traffic Classes 7

Criteria for Mapping Classes 8

Prerequisites for Classification 9

Guidelines and Limitations for QoS Classification 9

Classifying Traffic 9

Classifying ACL Traffic 9

Classifying DSCP Traffic 11

Configuring IP Precedence Classification 12

Configuring QoS Group Classification 13

Configuring Discard Class Classification 14

Configuring Layer 3 Packet Length Classification 15

Configuring CoS Classification 16

Configuring IP RTP Classification 17

Configuring Class Map Classification 18

Verifying the Classification Configuration 20

Configuration Example for QoS Classification 20

Feature History for QoS Classification 20

CHAPTER 3 **Configuring QoS Marking Policies** 21

Information About Policy Maps 21

Criteria for Marking Fields 22

Prerequisites for QoS Marking Policies 23

Guidelines and Limitations for QoS Marking Policies 23

Creating QoS Marking Policies 23

Creating a DSCP Policy 23

Creating an IP Precedence Policy 25

Creating a Class of Service Policy 26

Creating a QoS Group Policy 28

Creating a Discard Class Policy 29

Creating Ingress and Egress Policies 30

Verifying the QoS Policy Configuration 32

Configuration Example for QoS Marking Policies 32

Feature History for QoS Marking Policies 33

CHAPTER 4 **Configuring QoS Policing** 35

Information About Policing 35

Prerequisites for Policing 36

Guidelines and Limitations for QoS Policing 37

Configuring Ingress and Egress Policing 37

Configuring Policing 37

Police Command and Policer Types 37

Policer Action 38

Police Command Data Rates 40

Police Command Burst Sizes 40

Configuring Markdown Policing 40

Verifying the Policing Configuration 42

Configuration Example for QoS Policing 42

Feature History for QoS Policing 43

CHAPTER 5 **Monitoring QoS Statistics** 45

Information About QoS Statistics 45

Prerequisites for Monitoring QoS Statistics 45

Enabling QoS Statistics 46
Displaying QoS Statistics 47
Clearing QoS Statistics 49
Configuration Example for QoS Statistics 49
Standards 50
MIBs 50
RFCs 50
Technical Assistance 51
Feature History for QoS Statistics 51

CHAPTER 1

# Overview

This chapter contains the following sections:


- Information About Quality of Service, page 1
- Traffic Classification and Marking, page 2
- QoS Commands, page 2
- Default QoS Behavior, page 3
- Supported RFCs, page 3
- High Availability Requirements for QoS Features, page 4
- Commonly Used DSCP Values, page 4
- IP Precedence Values, page 5


## Information About Quality of Service

You can use quality of service (QoS) to provide the most desirable flow of traffic through a network. QoS allows you to classify your network traffic, police and prioritize the traffic flow, and provide congestion avoidance. Traffic is processed based on how you classify it and the QoS policies that you put in place.

You can implement a QoS policy using the following steps:

1. Define a traffic class by using the **class-map** command. For more information, see Configuring QoS Classification, on page 7.
2. Create a traffic class by using the **policy-map** command. A traffic policy defines how specific traffic is to be acted upon to improve the quality of service. For more information, see Configuring QoS Marking Policies, on page 21.
3. Attach the traffic policy to an interface or port profile by using the **service-policy** command. For more information, see Creating Ingress and Egress Policies, on page 30.
4. Police the traffic. For more information, see Configuring QoS Policing, on page 35.

# Traffic Classification and Marking

QoS classifies network traffic, uses or assigns QoS labels to indicate priority, makes the packets comply with the configured resource usage limits (polices the traffic and marks the traffic), and provides congestion avoidance where resource contention exists. The following table describes these processes.

| QoS Method | Description | Command | Mechanism |
|---|---|---|---|
| Traffic Classifications | Groups network traffic based on defined criteria. | **match** | class maps |
| Traffic Marking | Modifies traffic attributes by matching the class. | **set** | policy maps |

# QoS Commands

QoS configuration commands are shown in the following table.

| Command | Configuration | Description |
|---|---|---|
| **class-map** | Global configuration | Defines a class map that represents a class of traffic. |
| **table-map** | Global configuration | Defines a table map that represents a mapping from one set of field values to another set of field values. You can reference a table map from a policy map. |
| **policy-map** | Global configuration | Defines a policy map that represents a set of policies to be applied to a set of class maps. Policy maps can reference table maps. |
| **match** | Class map QoS configuration | Defines the criteria for a class map. |
| **set** | Policy map QoS configuration | Defines the action to be taken on the packet. |

| Command | Configuration | Description |
|---|---|---|
| **service-policy** | Interface or port profile configuration | Applies a specified policy map to input or output packets on interfaces configured as follows:<br>• inherited from a port-profile<br>• port-channel<br>• Ethernet<br>• vEthernet |
| **police** | Policy map class QoS configuration | Defines the rate at which data traffic is monitored. |

# Default QoS Behavior

QoS has no default behavior. Policing and prioritization of traffic are implemented only when you apply a policy map to an interface. When you are configuring QoS with an ACL, note that packets are processed as follows:

- QoS ingress processing follows ACL processing.
- QoS egress processing precedes ACL egress processing.

# Supported RFCs

The following table lists RFCs that are supported by QoS.

| Number | Title |
|---|---|
| RFC 2475 | Architecture for Differentiated Services |
| RFC 2697 | A Single Rate Three Color Marker |
| RFC 2698 | A Dual Rate Three Color Marker |
| RFC 3289 | Management Information Base for the Differentiated Services Architecture |
| RFC 3550 | RTP: A Transport Protocol for Real-Time Applications |

# High Availability Requirements for QoS Features

QoS recovers its previous state after a software restart, and it is able to switch over from the active supervisor to the standby supervisor without a loss of state.

# Commonly Used DSCP Values

You can mark both incoming and outgoing packets. The following commonly used DSCP values are described in RFC 2475.

| DSCP Value | Decimal Value | Meaning | Drop Probability | Equivalent IP Precedence Values |
|---|---|---|---|---|
| 101 110 | 46 | High Priority Expedited Forwarding (EF) | N/A | 101—Critical |
| 000 000 | 0 | Best effort | N/A | 000—Routine |
| 001 010 | 10 | AF11 | Low | 001—Priority |
| 001 100 | 12 | AF12 | Medium | 001—Priority |
| 001 110 | 14 | AF13 | High | 001—Priority |
| 010 010 | 18 | AF21 | Low | 010—Immediate |
| 010 100 | 20 | AF22 | Medium | 010—Immediate |
| 010 110 | 22 | AF23 | High | 010—Immediate |
| 011 010 | 26 | AF31 | Low | 011—Flash |
| 011 100 | 28 | AF32 | Medium | 011—Flash |
| 011 119 | 30 | AF33 | High | 011—Flash |
| 100 010 | 34 | AF41 | Low | 100—Flash Override |
| 100 100 | 36 | AF42 | Medium | 100—Flash Override |
| 100 110 | 38 | AF43 | High | 100—Flash Override |
| 001 000 | 8 | CS1 | | 1 |
| 010 000 | 16 | CS2 | | 2 |
| 011 000 | 24 | CS3 | | 3 |

| DSCP Value | Decimal Value | Meaning | Drop Probability | Equivalent IP Precedence Values |
|---|---|---|---|---|
| 100 000 | 32 | CS4 | | 4 |
| 101 000 | 40 | CS5 | | 5 |
| 110 000 | 48 | CS6 | | 6 |
| 111 000 | 56 | CS7 | | 7 |
| 000 000 | 0 | Default | | |
| 101 110 | 46 | EF | | |

# IP Precedence Values

The IP precedence values from least to most important are listed in the following table.

| Value | Description |
|---|---|
| 000 (0) | Routine or Best Effort |
| 001 (1) | Priority |
| 010 (2) | Immediate |
| 011 (3) | Flash (mainly used for voice signaling or for video) |
| 100 (4) | Flash Override |
| 101 (5) | Critical (mainly used for voice RTP) |
| 110 (6) | Internet |
| 111 (7) | Network |

CHAPTER 2

# Configuring QoS Classification

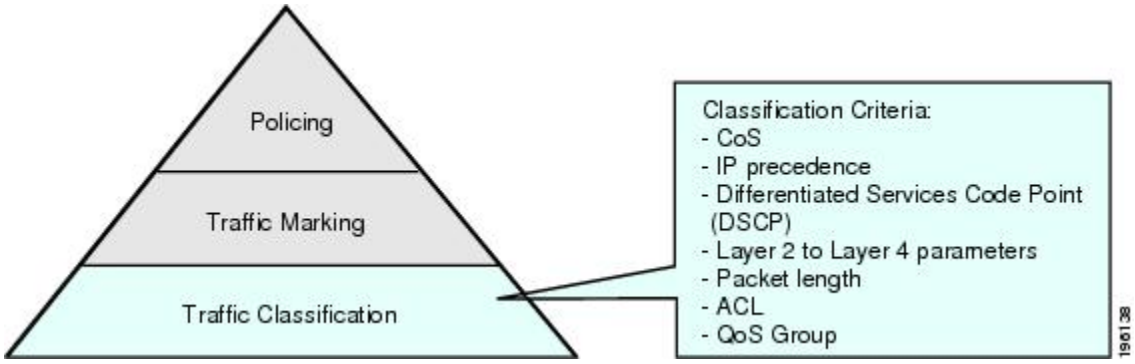
This chapter contains the following sections:


- Information About Traffic Classes, page 7
- Criteria for Mapping Classes, page 8
- Prerequisites for Classification, page 9
- Guidelines and Limitations for QoS Classification, page 9
- Classifying Traffic, page 9
- Verifying the Classification Configuration, page 20
- Configuration Example for QoS Classification, page 20
- Feature History for QoS Classification, page 20


## Information About Traffic Classes

Traffic classes, or categories of traffic (packets) that are grouped on the basis of similarity of traffic, are called class maps. Classifying network traffic allows you to enable a quality of service (QoS) strategy in your network.

***Figure 1: Criteria for Classifying Network Traffic***



Identifying and categorizing network traffic into traffic classes (that is, classifying packets) enables you to handle different types of traffic by separating network traffic into different categories.

Classifying network traffic allows you to see the kinds of traffic you have and treat some types of traffic differently than others. Identifying and organizing network traffic allows you to allocate network resources to deliver the best performance for each type of traffic.

You can place network traffic with a specific IP precedence into one traffic class, while you place traffic with a specific differentiated services code point (DSCP) value into another traffic class. Each traffic class can be given a different QoS class, which you configure in a policy map later.

You define each class of traffic in a class map based upon criteria, such as the IP precedence or class of service (CoS). The allowable criteria for mapping classes of traffic is available and you can match the criteria to your traffic as follows:

- Matching all
- Matching or not matching one
- Matching or not matching multiple
- Matching or not matching another class map

Some of the criteria used in traffic class maps relates only to one direction of traffic—either ingress or egress. For example, the internal label QoS group has no meaning on ingress traffic because it has not yet been assigned a value.

Traffic that fails to match any traffic class in a QoS policy map is assigned to a default class of traffic called class-default. The class-default can be referenced in a QoS policy map to select this unmatched traffic.

# Criteria for Mapping Classes

The following table lists the allowable criteria for the mapping traffic classes.

| Class Criteria | Description |
|---|---|
| CoS | Class of service (CoS) field in the IEEE 802.1Q header. |
| IP precedence | Precedence value within the type of service (ToS) byte of the IP header.<br>For details, see IP Precedence Values, on page 5. |
| Differentiated Services Code Point (DSCP) | DSCP value within the DIffServ field of the IP header. The standard DSCP values are listed in Commonly Used DSCP Values, on page 4. |
| QoS group | Locally significant QoS values that can be manipulated and matched within the system. The range is from 0 to 126. |
| Discard class | Locally significant values that can be matched and manipulated within the system. The range is from 0 to 63. |

| Class Criteria | Description |
|---|---|
| ACL | IP access control list (ACL) or MAC ACL name.<br>If you configure the class to match-all and ACL as match-criteria, no other match criteria, except the packet length, can be specified.<br>If you configure the class to match-any and ACL as match-criteria, you can also match ACLs and any other match criteria. |
| Packet length | Size range of Layer 3 packet lengths. |
| IP RTP | Applications that are using the Real-time Transport Protocol (RTP) are identified by UDP port number range. |
| Class map | Criteria that are specified in a named class-map object. |

# Prerequisites for Classification

You are logged in to the CLI in EXEC mode.

# Guidelines and Limitations for QoS Classification

- You can specify a maximum of 32 match criteria in a class map.
- You can configure a maximum of 64 classes for use in a single policy map if no policers are configured.

**Note** Tunneled IP packets are not matched unless the tunneling protocol is also IP, and then the match applies to the outer IP header and not the encapsulated IP header.

# Classifying Traffic

## Classifying ACL Traffic

You can classify traffic by matching packets based on existing access control lists (ACLs).

### Before You Begin

- QoS does not use the permit-deny functions of ACLs. The **permit** and **deny** ACL keywords are ignored when matching.
- QoS does not support the **not** form of this command.

- If you configure the class to match-all and ACL as match-criteria, no other match criteria, except the packet length, can be specified.
- If you configure the class to match-any and ACL as match-criteria, you can match ACLs and any other match criteria also.
- You are logged in to the CLI in EXEC mode.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-cmap-qos)# **match access-group name** *ac1-name* | Configures and saves the access group to match for this class in the running configuration.<br>**Note** The permit and deny keywords are ignored when matching the ACL.<br>The **not** form of this command is not supported. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to classify the ACL traffic:

```
switch# configure terminal
switch(config)# class-map class_acl
switch(config-cmap-qos)# match access-group name my_acl
switch(config-cmap-qos)# show class-map class_acl
 Type qos class-maps
 ===================

  class-map type qos match-all class_acl
  match access-group name my_acl

switch(config-cmap-qos)# copy running-config startup-config
```

# Classifying DSCP Traffic

You can classify traffic based on the DSCP value in the DiffServ field of the IP header. The standard DSCP values are found in the Commonly Used DSCP Values, on page 4 and IP Precedence Values, on page 5.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-cmap-qos)# **match [not] dscp** *dscp_list* | Configures the traffic class by matching packets that are based on *dscp-values*. The standard DSCP values are listed in the Commonly Used DSCP Values, on page 4 and IP Precedence Values, on page 5.<br>Use the **not** keyword to match on values that do not match the specified range. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to classify DSCP traffic:

```
switch# configure terminal
switch(config)# class-map class_dscp
switch(config-cmap-qos)# match dscp af21 af32

 Type qos class-maps
 ===================

   class-map type qos match-all class_dscp
     match dscp 18,28
switch(config-cmap-qos)# show class-map class_dscp
switch(config-cmap-qos)# copy running-config startup-config
```

# Configuring IP Precedence Classification

You can classify traffic based on the precedence value in the type of service (ToS) byte field of the IP header. Precedence values can be found in the Commonly Used DSCP Values, on page 4 and IP Precedence Values, on page 5 tables.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-cmap-qos)# **match [not] precedence** *values* | Configures the traffic class by matching packets that are based on *precedence-values* listed in the following table:<br>**Value** / **Description**<br>000 (0) — Routine or Best Effort<br>001 (1) — Priority<br>010 (2) — Immediate<br>011 (3) — Flash (mainly used for voice signaling or for video)<br>100 (4) — Flash Override<br>101 (5) — Critical (mainly used for voice RTP)<br>110 (6) — Internet<br>111 (7) — Network<br>Use the **not** keyword to match on values that do not match the specified range. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure the IP precedence classification:

```
switch# configure terminal
switch(config)# class-map class_ip_precedence
switch(config-cmap-qos)# match precedence 1-2, 5-7
switch(config-cmap-qos)# show class-map class_ip_precedence
 Type qos class-maps
 ====================

   class-map type qos match-all class_ip_precedence
     match precedence 1-2,5-7

switch(config-cmap-qos)# copy running-config startup-config
```

# Configuring QoS Group Classification

You can classify traffic based on the QoS group internal label, which is not part of the packet payload or any packet header. You can set the value of the QoS group within a policy map using the **set qos-group** command as described in the Creating a QoS Group Policy, on page 28.

**Note** You match on the QoS group only in egress policies because its value is undefined until you set it in an ingress policy.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-cmap-qos)# **match [not] qos-group** *multi-range-qos-group-values* | Configures the traffic class by matching packets that are based on a list of QoS group values. Values can range from 0 to 126. The default QoS group value is 0. Use the **not** keyword to match on values that do not match the specified range. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration for the specified traffic class name. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure QoS group classification:

```
switch# configure terminal
switch(config)# class-map class_qos_group
switch(config-cmap-qos)# match qos-group 4, 80-90
switch(config-cmap-qos)# show class-map class_qos_group

 Type qos class-maps
 ===================

   class-map type qos match-all class_qos_group
     match qos-group 4,80-90

switch(config-cmap-qos)# copy running-config startup-config
```

# Configuring Discard Class Classification

You can classify traffic based on the value of the discard class internal label, which is not part of the packet payload or any packet header. You can set the value of the discard class within a policy map by using the **set discard-class** command.

**Note** You match on the discard class only in egress policies because its value is undefined until you set it in an ingress policy.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-cmap-qos)# **match [not] discard-class** *multi-range-discard-group-values* | Configures the traffic class by matching packets that are based on a list of discard-class values. Values can range from 0 to 63. The default discard class value is 0. Use the **not** keyword to match on values that do not match the specified range. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration for the specified traffic class name. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure discard class classification:

```
switch# configure terminal
switch(config)# class-map class_discard_class
switch(config-cmap-qos)# match discard-class 4, 60-62
switch(config-cmap-qos)# show class-map class-discard-class
 Type qos class-maps
 ====================

   class-map type qos match-all class_dicard_class
     match discard-class 4,60-62
switch(config-cmap-qos)# copy running-config startup-config
```

# Configuring Layer 3 Packet Length Classification

You can classify Layer 3 traffic based on various packet lengths.

**Note** This feature is designed for IP packets only.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-cmap-qos)# **match [not] packet-length** *packet-length-list* | Configures the traffic class by matching packets that are based on various packet lengths. Values can range from 1 to 9198. Use the **not** keyword to match on values that do not match the specified range. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration for the specified traffic class name. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure Layer 3 packet length classification:

```
switch# configure terminal
switch(config)# class-map class_packet_length
switch(config-cmap-qos)# match packet length 2000
switch(config-cmap-qos)# show class-map class_packet_length
 Type qos class-maps
 ===================
   class-map type qos match-all class_packet_length
     match packet length 2000
switch(config-cmap-qos)# copy running-config startup-config
```

# Configuring CoS Classification

Traffic classification allows you to organize traffic (packets) into traffic classes or categories on the basis of whether the traffic matches the criteria you that specify. The values used to classify traffic are called match criteria. When you define a traffic class, you can specify multiple match criteria, you can choose to not match on a particular criterion, or you can determine the traffic class by matching any or all criteria.

You can classify traffic based on the class of service (CoS) in the IEEE 1Q header. This 3-bit field is defined in IEEE.802.1p to support QoS traffic classes. CoS refers to three bits in a 802.1Q header that is used to indicate the priority of the Ethernet frame as it passes through a switch network.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-cmap-qos)# **match** [**not**] **cos** *cos-list* | Configures the traffic class by matching packets that are based on a list of CoS values. Values can range from 0 to 7. Use the **not** keyword to match on values that do not match the specified range. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration for the specified traffic class name. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure CoS classification:

```
switch# configure terminal
switch(config)# class-map class_cos
switch(config-cmap-qos)# match cos 4, 5-6
switch(config-cmap-qos)# show class-map class-cos
 Type qos class-maps
 ====================

   class-map type qos match-all class_cos
     match cos 4-6

switch(config-cmap-qos)# copy running-config startup-config
```

# Configuring IP RTP Classification

The IP Real-time Transport Protocol (RTP) is a transport protocol for real-time applications that transmits data such as audio or video and is defined by RFC 3550. Although RTP does not use a common TCP or UDP port, you typically configure RTP to use ports 16384 to 32767. UDP communications use an even port and the next higher odd port is used for RTP Control Protocol (RTCP) communications.

You can configure classification based on UDP port ranges, which are likely to target applications using RTP.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-cmap-qos)# **match** [**not**] **ip rtp** *udp-port-values* | Configures the traffic class by matching packets that are based on the range of lower and upper UDP port numbers, which is likely to target applications using RTP. Values can range from 2000 to 65535. Use the **not** keyword to match on values that do not match the specified range. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration for the specified traffic class name. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure IP RTP classification:

```
switch# configure terminal
switch(config)# class-map class_rtp
switch(config-cmap-qos)# match ip rtp 2000-21000, 4000-4100
 Type qos class-maps
 ===================

   class-map type qos match-all class_rtp
     match ip rtp 2000-2100,4000-4100
switch(config-cmap-qos)# show class-map class-rtp
switch(config-cmap-qos)# copy running-config startup-config
```

# Configuring Class Map Classification

You can classify traffic based on the match criteria in another class map.

### Before You Begin

- The referenced class map must be created prior to its reference.
- You can reference the same class map in multiple policies.
- You can configure only one level of nesting of class maps. You cannot reference a class map that references another class map.
- Before you delete a referenced class map, you should delete all references to that class map.
- To perform a logical OR with the class map that is specified in the **match class-map** command, use the **match-any** keyword. The **match-any** or **match-all** specification of the matched class map is ignored.
- To perform a logical AND with the class map that is specified in the **match class-map** command, use the **match-all** keyword. The **match-any** or **match-all** specification of the matched class map is ignored.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **class-map** [**type qos**] [**match-any** \| **match-all**] *class_map_name* | Places you into class map QoS configuration mode for the specified class map and configures and saves the map name in the running configuration.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-cmap-qos)# **match** [**not**] **class-map** *class_map_name* | Configures the traffic class by matching packets that are based on the match criteria in another class map. Because **match-all** is the default for the **class-map** command, the match criteria that is specified in class_map3 are ANDed with match criteria in class_class_map. Use the **not** keyword to find values that do not match the specified range. |
| **Step 4** | switch(config-cmap-qos)# **show class-map** *class_map_name* | Displays the class map configuration. |
| **Step 5** | switch(config-cmap-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure the class map classification:

```
switch# configure terminal
switch(config)# class-map class_class_map
switch(config-cmap-qos)# match class-map class-map
switch(config-cmap-qos)# show class-map class_class_map

 Type qos class-maps
 ===================

   class-map type qos match-all class_class_map
     match class-map class-map

switch(config-cmap-qos)# show class-map class-map


 Type qos class-maps
 ===================

   class-map type qos match-all class-map
     match dscp 10,12

switch(config-cmap-qos)# copy running-config startup-config
```

# Verifying the Classification Configuration

To verify the classification configuration, use the commands in the following table.

| Command | Description |
|---|---|
| **show class-map** *name* | Displays the class map configuration for all class maps or for a specified class map. |
| **show ip access-lists** *name* | Displays all IPv4 access control lists (ACLs) or a specific IPv4 ACL. |

# Configuration Example for QoS Classification

This example shows how to configure classification for the class map named cmap1, which matches DSCP traffic AF21 and AF32:

```
switch(config)# class-map type qos match-all cmap1
switch(config-cmap-qos)# show class-map cmap1
 Type qos class-maps
 ===================

   class-map type qos match-all cmap1
     match dscp 18,28


switch(config-cmap-qos)# exit
```

# Feature History for QoS Classification

This section provides the QoS classification release history.

| Feature Name | Release | Feature Information |
|---|---|---|
| QoS Classification | 5.2(1)SM1(5.1) | This feature was introduced |

CHAPTER 3

# Configuring QoS Marking Policies

This chapter contains the following sections:


- Information About Policy Maps, page 21
- Criteria for Marking Fields, page 22
- Prerequisites for QoS Marking Policies, page 23
- Guidelines and Limitations for QoS Marking Policies, page 23
- Creating QoS Marking Policies, page 23
- Verifying the QoS Policy Configuration, page 32
- Configuration Example for QoS Marking Policies, page 32
- Feature History for QoS Marking Policies, page 33


## Information About Policy Maps

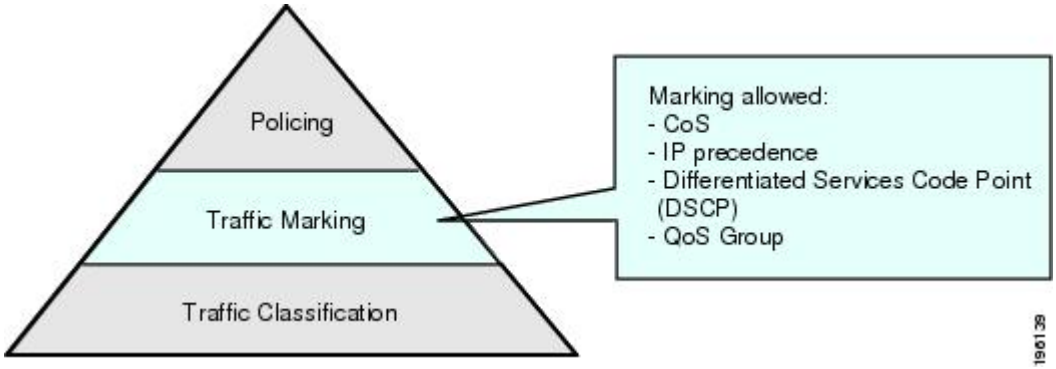
Policy maps prioritize network traffic by class. You create policy maps to define how to treat each class of traffic so that it is prioritized for the best quality of service.

Marking is a method that you use to modify the QoS fields of the incoming and outgoing packets. The QoS fields that you can mark are CoS in Layer 2, and IP precedence and Differentiated Service Code Point (DSCP) in Layer 3. The QoS group and discard class are two labels local to the system that you can assign intermediate marking values. You can use these two labels to determine the final values marked in a packet.

Service policies are specified using policy maps. Policy maps provide an ordered mapping of class maps to service levels. You can specify multiple class maps within a policy map, and map a class map to a high, medium, or low service level. The default priority is low. The policy map name is restricted to 63 alphanumeric characters.

The order of the class maps within a policy map is important to determine the order in which the frame is compared to class maps. The first matching class map has the corresponding priority marked in the frame.

***Figure 2: Packet Fields Available for Marking***




After you define your traffic classes, you can reference them in the policy map where you also define how they should be marked.

# Criteria for Marking Fields

The following table lists fields that are available for marking.

| Field | Description |
|---|---|
| DSCP | Layer 3 differentiated services code point (DSCP).<br>**Note** If you manipulate this DSCP value, you cannot manipulate discard class values, and vice-versa. |
| IP Precedence | Layer 3 IP precedence.<br>**Note** IP precedence uses only the lower 3 bits of the type of service (ToS) field. The device overwrites the first 3 bits of the ToS field to 0. |
| CoS | Layer 2 class of service (CoS). |
| QoS Group | Local QoS values that can be marked and matched as needed. The range is from 0 to 126. |
| Discard Class | Local QoS values that can be matched and marked as needed. The range is from 0 to 63.<br>**Note** If you manipulate this discard class value, you cannot manipulate DSCP values, and vice-versa. |
| Using Table Maps | Method to use table maps for marking. |

Unless noted as a restriction, you can mark both incoming and outgoing packets.

# Prerequisites for QoS Marking Policies

Marking has the following prerequisites:

- You must have already classified your network traffic. For more information, see Configuring QoS Classification, on page 7.
- You are already logged in to the CLI in EXEC mode.

# Guidelines and Limitations for QoS Marking Policies

- The **set cos** command is applicable only to 802.1Q interfaces. So, although you can use the **set cos** command on an ingress interface, the setting is only applied if a packet eventually egresses an 802.1Q compliant interface.
- For a single class, you can set operations on any two out of the following five fields: CoS, IP Precedence, DSCP, QoS Group, and Discard Class.
- You can use the **set qos-group** command only in ingress policies.
- You can use the **set discard-class** command only in ingress policies.

# Creating QoS Marking Policies

## Creating a DSCP Policy

You can create a policy that marks the DSCP value in the IP header packet to prioritize traffic in a particular class.

### Before You Begin

- See the DSCP description in RFC 2475.
- You are logged in to the CLI in EXEC mode.
- If you use DSCP marking, you cannot use Discard Class marking.
- You can mark the DSCP field as a numeric value between 0 and 63 or as one of the commonly used values listed in Commonly Used DSCP Values, on page 4 and IP Precedence Values, on page 5.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | switch(config)# **policy-map** [**type qos**] [**match-first**] *policy-map-name* | Places you into policy map QoS configuration mode for the specified policy map and configures the map name in the running configuration.<br>The *policy-map-name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-pmap)# **class** [**type qos**] {*class_map_name* \|**class-default**} | Creates a reference to *class-map-name* and enters policy-map class QoS configuration mode for the specified class map. By default, the class is added to the end of the policy map. Changes are saved in the running configuration.<br>Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-insensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 4** | switch (config-pmap-c-qos)# **set dscp** *value* | Defines the DSCP value that should be used in all IP headers for the specified class and saves it in the running configuration.<br>You can use a numeric value from 1 to 60 or one of the standard values from Commonly Used DSCP Values, on page 4 and IP Precedence Values, on page 5. |
| **Step 5** | switch(config-pmap-c-qos)# **show policy-map** *policy-map-name* | Displays the policy map configuration for the specified map name. |
| **Step 6** | switch(config-pmap-c-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to create a DSCP policy:

```
switch# configure terminal
switch(config)# policy-map policy1
switch(config-pmap)# class class1
switch(config-pmap-c-qos)# set dscp af31
switch(config-pmap-c-qos)# show policy-map policy1
switch(config-pmap-c-qos)# show policy-map policy
Type qos policy-maps
 ====================

 policy-map type qos policy1
   class  class1
     set dscp af31
```

```
switch(config-pmap-c-qos)# copy running-config startup-config
```

# Creating an IP Precedence Policy

### Before You Begin

- You are logged in to the CLI in EXEC mode.
- See the RFC 791 precedence values from least to most important in the table for IP Precedence Values, on page 5.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **policy-map** [**type qos**] [**match-first**] *policy-map-name* | Places you into policy map QoS configuration mode for the specified policy map and configures the map name in the running configuration.<br>The *policy-map-name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-pmap-qos)# **class** [**type qos**] {*class_map_name* \|**class-default**} | Creates a reference to *class-map-name* and enters policy-map class QoS configuration mode for the specified class map. By default, the class is added to the end of the policy map. Changes are saved in the running configuration.<br>Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 4** | switch (config-pmap-c-qos)# **set precedence** *value* | Adds the precedence value that should be used in all IP headers for the specified class and saves it in the running configuration.<br>You can use a numeric value from 0 to 7 as shown in the following table: |

| Value | Description |
|---|---|
| 000 (0) | Routine or Best Effort |
| 001 (1) | Priority |

| | Command or Action | Purpose |
|---|---|---|
| | | <table><tr><th>Value</th><th>Description</th></tr><tr><td>010 (2)</td><td>Immediate</td></tr><tr><td>011 (3)</td><td>Flash (mainly used for voice signaling or for video)</td></tr><tr><td>100 (4)</td><td>Flash Override</td></tr><tr><td>101 (5)</td><td>Critical (mainly used for voice RTP)</td></tr><tr><td>110 (6)</td><td>Internet</td></tr><tr><td>111 (7)</td><td>Network</td></tr></table> |
| **Step 5** | switch(config-pmap-c-qos)# **show policy-map** *policy-map-name* | Displays the policy map configuration for the specified map name. |
| **Step 6** | switch(config-pmap-c-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to create an IP precedence policy:

```
switch# configure terminal
switch(config)# policy-map policy1
switch(config-pmap)# class class1
switch(config-pmap-c-qos)# set precedence 3
switch(config-pmap-c-qos)# show policy-map policy1
 Type qos policy-maps
 ====================

 policy-map type qos policy1
   class  class1
     set prec 3
switch(config-pmap-c-qos)# copy running-config startup-config
```

# Creating a Class of Service Policy

You can mark the CoS field in the IEEE 802.1Q header for all traffic in a specific class. If you mark this field in an ingress or egress policy, it will only be set when a packet egresses an IEEE 802.1Q-capable interface.

### Before You Begin

- You are logged in to the CLI in EXEC mode.
- You can set CoS in ingress and egress policies.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **policy-map** [**type qos**] [**match-first**] *policy-map-name* | Places you into policy map QoS configuration mode for the specified policy map and configures the map name in the running configuration.<br>The *policy-map-name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-pmap-qos)# **class** [**type qos**] {*class_map_name* \| **class-default**} | Creates a reference to *class-map-name* and enters policy-map class QoS configuration mode for the specified class map. By default, the class is added to the end of the policy map. Changes are saved in the running configuration.<br>Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 4** | switch (config-pmap-c-qos)# **set cos** *cos-value* | Sets the CoS value to *cos-value*. The value can range from 0 to 7. You can use this command only in egress policies. |
| **Step 5** | switch(config-pmap-c-qos)# **show policy-map** *policy-map-name* | Displays the policy map configuration for the specified map name. |
| **Step 6** | switch(config-pmap-c-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to create a class of service policy:

```
switch# configure terminal
switch(config)# policy-map policy1
switch(config-pmap-qos)# class class1
switch(config-pmap-c-qos)# set cos 3
switch(config-pmap-c-qos)# show policy-map policy1
 Type qos policy-maps
 ====================

 policy-map type qos policy1
   class  class1
```

```
     set cos 3
switch(config-pmap-c-qos)#
```

# Creating a QoS Group Policy

### Before You Begin

- You are logged in to the CLI in EXEC mode.
- You can mark the QoS group value only in ingress policies.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **policy-map** [**type qos**] [**match-first**] *policy-map-name* | Places you into policy map QoS configuration mode for the specified policy map and configures the map name in the running configuration.<br>The *policy-map-name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-pmap-qos)# **class** [**type qos**] {*class_map_name* \| **class-default**} | Creates a reference to *class-map-name* and enters policy-map class QoS configuration mode for the specified class map. By default, the class is added to the end of the policy map. Changes are saved in the running configuration.<br>Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 4** | switch (config-pmap-c-qos)# **set qos-group** *qos-group-value* | Sets the QoS group value to qos-group-value. The value can range from 0 to 126. |
| **Step 5** | switch(config-pmap-c-qos)# **show policy-map** *policy-map-name* | Displays the policy map configuration for the specified map name. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | switch(config-pmap-c-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to create a QoS group policy:

```
switch# configure terminal
switch(config)# policy-map policy1
switch(config-pmap-qos)# class class1
switch(config-pmap-c-qos)# set qos-group 100
switch(config-pmap-c-qos)# show policy-map policy1
 Type qos policy-maps
 ====================

 policy-map type qos policy1
   class  class1
     set qos-group 100


switch(config-pmap-c-qos)# copy running-config startup-config
```

# Creating a Discard Class Policy

You can set a local internal label discard class policy.

### Before You Begin

- If you configure a local internal label discard class policy, you cannot create a DSCP policy. For more information about DSCP policies, see Creating a DSCP Policy, on page 23.
- You can set a discard class only in ingress policies.
- To reference the local discard class in a policy or in traffic classification, use the **match discard-class** command. For more information, see Configuring Discard Class Classification, on page 14.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **policy-map** [**type qos**] [**match-first**] *policy-map-name* | Places you into policy map QoS configuration mode for the specified policy map and configures the map name in the running configuration.<br>The *policy-map-name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-pmap-qos)# **class** [**type qos**] {*class_map_name* \|**class-default**} | Creates a reference to *class-map-name* and enters policy-map class QoS configuration mode for the specified class map. By default, the class is added to the end of the policy map. Changes are saved in the running configuration.<br>Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 4** | switch (config-pmap-c-qos)# **set discard-class** *discard-class-value* | Sets the discard-class value to *discard-class-value*. The value ranges from 0 to 63. |
| **Step 5** | switch(config-pmap-c-qos)# **show policy-map** *policy-map-name* | Displays the policy map configuration for the specified map name. |
| **Step 6** | switch(config-pmap-c-qos)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to create a discard class policy:

```
switch# configure terminal
switch(config)# policy-map policy1
switch(config-pmap-qos)# class class1
switch(config-pmap-c-qos)# set discard-class 40
switch(config-pmap-c-qos)# show policy-map policy1
 Type qos policy-maps
 ====================
 policy-map type qos policy1
   class  class1
     set discard-class 40

switch(config-pmap-c-qos)# copy running-config startup-config
```

# Creating Ingress and Egress Policies

You can attach a policy map to an interface or a port profile so that the marking instructions are applied to the ingress or egress packets.

### Before You Begin

- The interface or port profile have been created.
- The policy map that you want to use has been defined.

**Note** You can attach only one input policy and one output policy to an interface or port profile.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | Enter one of the following commands:<br>• switch (config)# **interface** *type number*<br>• switch (config)# **port-profile** *name* | Places you into configuration mode for the specified Ethernet or vEthernet interface or port profile. |
| **Step 3** | switch (config-if)# **service-policy** [**type qos**] {**input** \| **output**} *policy-map-name* [**no stats**] | (Optional) Attaches a policy map name that will be added to the input or output packets of the interface or port profile.<br>**Note** You can attach only one input policy and one output policy to an interface or port profile.<br>The *policy-map-name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 4** | switch(config-if)# **show policy-map** *policy-map-name* | Displays the policy map configuration for the specified map name. |
| **Step 5** | switch(config-if)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

This example shows how to configure an ingress policy on an Ethernet interface:

```
switch# configure terminal
switch(config)# interface port-channel 1
switch(config-if)# service-policy input policy1
switch(config-if)# show policy-map interface port-channel 1 input

Global statistics status :   enabled

port-channel1

 Service-policy (qos) input:   policy1
   policy statistics status:   enabled

   Class-map (qos):   class1 (match-all)
     283 packets
     Match: access-group testacl1
     Match: packet length 3-6000
```

```
     set dscp af31
switch(config-if)# copy running-config startup-config
```

This example shows how to configure an egress policy on a port profile:

```
switch# configure terminal
switch(config)# port-profile 2225-ephe
switch(config-port-prof)# service-policy output policy1
switch(config)# show policy-map interface vethernet 1
Global statistics status :   enabled

Vethernet1

 Service-policy (qos) output:   policy1
   policy statistics status:   enabled

   Class-map (qos):   class1 (match-all)
     0 packets
     Match: access-group testacl1
     Match: packet length 3-6000
     set dscp af31

switch(config)# copy running-config startup-config
```

# Verifying the QoS Policy Configuration

Use one of the following commands to verify the configuration:

| Command | Description |
|---|---|
| **show policy-map** [**type qos**] [**name** *policy_map_name*] | Displays the policy map configuration. |
| **show policy-map interface** *name* | Displays the policy applied on an interface along with statistics. |
| **show policy-map interface brief** | Displays all the interfaces on which the QoS policy is applied. |
| **show running-config ipqos** | Displays all configured class and policy maps and interfaces on which they are being applied. |

# Configuration Example for QoS Marking Policies

This example shows how to display a specific policy-map policy applied on an interface:

```
switch(config)# show policy-map policy1
  Type qos policy-maps
  ====================

  policy-map type qos policy1
    class  class1
      set dscp af31
switch(config)# sh class-map class1
  Type qos class-maps
  ====================

    class-map type qos match-all class1
      match access-group name testacl1
      match packet length 3-6000

switch(config)# int veth 1
```

```
switch(config-if)# service-policy ty qos input policy1
switch(config-if)# sh run int veth 1

!Command: show running-config interface Vethernet1
!Time: Tue Feb 12 02:45:20 2013

version 5.2(1)SM1(5.1)

interface Vethernet1
  inherit port-profile 2225-ephe
  service-policy type qos input policy1
  description data-2-00-1, Network Adapter 2

switch(config-if)# sh policy-map interface vethernet 1
Global statistics status :   enabled

Vethernet1

  Service-policy (qos) input:   policy1
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      0 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      set dscp af31

switch(config-if)# sh policy-map interface brief

 Interface/VLAN [Status]:INP QOS       OUT QOS       INP QUE       OUT QUE
================================================================================
Vethernet1      [Active]:policy1
Vethernet2      [Active]:              policy1
Vethernet3      [Active]:              policy1
```

# Feature History for QoS Marking Policies

This section provides the QoS marking policies release history.

| Feature Name | Release | Feature Information |
|---|---|---|
| QoS Marking Policies | 5.2(1)SM1(5.1) | This feature was introduced. |

CHAPTER 4

# Configuring QoS Policing

This chapter contains the following sections:


- Information About Policing, page 35
- Prerequisites for Policing, page 36
- Guidelines and Limitations for QoS Policing, page 37
- Configuring Ingress and Egress Policing, page 37
- Configuring Policing, page 37
- Verifying the Policing Configuration, page 42
- Configuration Example for QoS Policing, page 42
- Feature History for QoS Policing, page 43


## Information About Policing

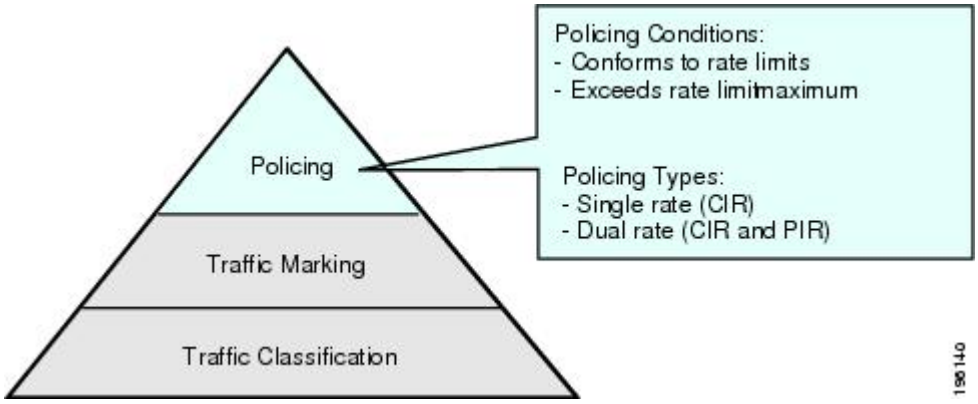
Policing is the monitoring of the data rates for a particular class of traffic. When the data rate exceeds user-configured values, marking or dropping of packets occurs immediately. Policing does not buffer the traffic, so transmission delay is not affected. When traffic exceeds the data rate, you instruct the system to either drop the packets or mark QoS fields in them. You can define single-rate, dual-rate, and color-aware policers.

Single-rate policers monitor the committed information rate (CIR) of traffic. Dual-rate policers monitor both the CIR and peak information rate (PIR) of traffic. In addition, the system monitors associated burst sizes. Three colors or conditions are determined by the policer for each packet depending on the data rate parameters supplied: conform (green), exceed (yellow), or violate (red).

You can configure only one action for each condition. For example, you might police for traffic in a class to conform to the data rate of 256,000 bits per second, with up to 200 millisecond bursts. The system would apply the conform action to traffic that falls within this rate, and it would apply the violate action to traffic that exceeds this rate.

Color-aware policers assume that traffic has been previously marked with a color. This information is then used in the actions taken by this type of policer. For more information about policies, see RFC 2697, RFC 2698, and RFC4115.

***Figure 3: Policing Condition and Types***




The following table lists the conditions that trigger actions by the policer depending on the defined data rate.

***Table 1: Policer Actions for Exceed or Violate***

| Condition | Color | Description | Policer Action[1] |
|---|---|---|---|
| Conform | Green | The packet traffic data rate is within the defined boundaries. | The policer either transmits these packets as is or changes the value in the header (DSCP, precedence, or CoS) and then transmits these packets. |
| Exceed | Yellow | The packet traffic data rate exceeds the defined boundary. | The policer can drop or mark down these packets. |
| Violate | Red | The packet traffic data rate violates the defined boundaries. | The policer can drop or mark down these packets. |

1 Only one policer action is allowed per condition.

# Prerequisites for Policing

- You are logged on to the CLI in EXEC mode.

# Guidelines and Limitations for QoS Policing

Each module polices independently, which might affect a policer that is applied to traffic distributed across more than one module, such as in the case of a port channel interface.

# Configuring Ingress and Egress Policing

You can apply the policing instructions in a QoS policy map to ingress or egress packets by attaching that QoS policy map to an interface or port profile. To select ingress or egress, you specify either the **input** or **output** keyword in the **service-policy** command. For an example of how to use the **service-policy** command, see Creating Ingress and Egress Policies, on page 30.

# Configuring Policing

## Police Command and Policer Types

### Police Command Arguments

The type of policer that is created by the Cisco Nexus 1000V is based on a combination of the **police** command arguments listed in the following table.

**Note** Specify the identical value for **pir** and **cir** to configure 1-rate, 3-color policing.

| Argument | Description |
|---|---|
| **cir** | Committed information rate (**cir**), or desired bandwidth, specified as a bit rate or a percentage of the link rate. Although a value for **cir** is required, the argument itself is optional. The range of values is from 1 to 80000000000; the range of mathematically significant policing values is 250 kbps to 80 Gbps. |
| **percent** | Rate as a percentage of the interface rate. The range of values is from 1 to 100%. |
| **bc** | Indication of how much the **cir** can be exceeded, either as a bit rate or an amount of time at **cir**. The default is 200 milliseconds of traffic at the configured rate. The default data rate units are bytes, and the Gigabit per second (gbps) rate is not supported for this parameter. |
| **pir** | Peak information rate (**pir**), which is specified as a PIR bit rate or a percentage of the link rate. There is no default. The range of values is from 1 to 80000000000; the range of mathematically significant policing values is from 250 kbps to 80 Gbps. The range of percentage values is from 1 to 100%. |

| Argument | Description |
|---|---|
| **be** | Indication of how much the **pir** can be exceeded, either as a bit rate or an amount of time at **pir**. When the **bc** value is not specified, the default is 200 milliseconds of traffic at the configured rate. The default data rate units are bytes, and the Gigabit per second (gbps) rate is not supported for this parameter.<br>**Note** You must specify a value for **pir** before the device displays this argument. |
| **conform** | Single action to take if the traffic data rate is within bounds. The basic actions are transmit or one of the **set** commands listed in the table. The default is transmit. |
| **exceed** | Single action to take if the traffic data rate exceeds the specified boundaries. The basic actions are drop or markdown. The default is drop. |
| **violate** | Single action to take if the traffic data rate violates the configured rate values. The basic actions are drop or markdown. The default is drop. |

### Policer Types and Actions

Although all the Police Command arguments are optional, you must specify a value for **cir**. In the following table, **cir** indicates the value but not necessarily the keyword itself. The combination of these arguments and the resulting policer types and actions are described.

| Police Arguments Present | Policer Type | Policer Action |
|---|---|---|
| **cir**, but not **pir**, **be**, or **violate** | 1-rate, 2-color | **≤ cir**, then **conform**; otherwise **violate** |
| **cir** and **pir** | 1-rate, 3-color | **≤ cir**, **conform**; **≤ pir**, **exceed**; else **violate**<br>**Note** You must specify identical values for **cir** and **pir**. |
| **cir** and **pir** | 2-rate, 3-color | **≤ cir**, then **conform**; **≤ pir**, then **exceed**; otherwise **violate** |

# Policer Action

You can take the actions listed in the following table when the packet exceeds the parameters or violates the parameters.

*Table 2: Policer Action for Exceed or Violate*

| Action | Description |
|---|---|
| **drop** | Drops the packet. This action is available only when the packet exceeds or violates the parameters. |
| **set dscp dscp table** {*cir-markdown-map* \| *pir-markdown-map*} | Sets the specified fields from a table map and transmits the packet. For more information on the system-defined, or default table maps, see Configuring QoS Marking Policies, on page 21. This action is available only when the packet exceeds the parameters (use the *cir-markdown-map*) or violates the parameters (use the *pir-markdown-map*). |

*Table 3: Policer Action for Conform*

| Action | Description |
|---|---|
| **transmit** | Transmits the packet. This action is available only when the packet conforms to the parameters. |
| **set-prec-transmit** | Sets the IP precedence field to a specified value and transmits the packet. This action is available only when the packet conforms to the parameters. |
| **set-dscp-transmit** | Sets the DSCP field to a specified value and transmits the packet. This action is available only when the packet conforms to the parameters. |
| **set-cos-transmit** | Sets the CoS field to a specified value and transmits the packet. This action is available only when the packet conforms to the parameters. |
| **set-qos-transmit** | Sets the QoS group internal label to a specified value and transmits the packet. This action can be used only in input policies and is available only when the packet conforms to the parameters. |
| **set-discard-class-transmit** | Sets the discard-class internal label to a specified value and transmits the packet. This action can be used only in ingress policies and is available only when the packet conforms to the parameters. |

# Police Command Data Rates

The policer can only drop or mark down packets that exceed or violate the specified parameters. For more information about marking down packets, see Configuring QoS Marking Policies, on page 21.

The **police** command uses the data rates listed in the following table.

***Table 4: Data Rates for the police Command***

| Rate | Description |
|---|---|
| bps | Bits per second (default) |
| kbps | 1000 bits per seconds |
| mbps | 1,000,000 bits per second |
| gbps | 1,000,000,000 bits per second |

# Police Command Burst Sizes

The **police** command uses the burst sizes in the following table.

| Speed | Description |
|---|---|
| bytes | bytes |
| kbytes | 1000 bytes |
| mbytes | 1,000,000 bytes |
| ms | milliseconds |
| us | microseconds |

# Configuring Markdown Policing

Markdown policing is the setting of a QoS field in a packet when traffic exceeds or violates the policed data rates. You can configure markdown policing by using the **set** commands.

**Note** You must specify the identical value for **pir** and **cir** to configure 1-rate, 3-color policing.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **policy-map** [**type qos**] [**match-first**] *policy-map-name* | Places you into policy map QoS configuration mode for the specified policy map and configures the map name in the running configuration.<br>The *policy-map-name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 3** | switch(config-pmap-qos)# **class** [**type qos**] {*class_map_name* \|**class-default**} | Creates a reference to *class-map-name* and enters policy-map class QoS configuration mode for the specified class map. By default, the class is added to the end of the policy map. Changes are saved in the running configuration.<br>Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.<br>The *class_map_name* argument is a unique alphabetic string that can be up to 40 case-sensitive characters long, including hyphen (-) and underscore (_) characters. The map name must be unique across class-maps and policy-maps. For example, you cannot have a class-map and a policy-map with the same name of HR_Map. |
| **Step 4** | switch(config-pamp-c-qos)# **police** [**cir**] {*committed-rate* [*data-rate*] \| **percent** *cir-link-percent*} [**bc** *committed-burst-rate* [*link-speed*]][**pir**] {*peak-rate* [*data-rate*] \| **percent** *cir-link-percent*} [**be** *peak-burst-rate* [*link-speed*]] [**conform** {**transmit** \| **set-prec-transmit** \| **set-dscp-transmit** \| **set-cos-transmit** \| **set-qos-transmit** \| **set-discard-class-transmit**} [**exceed** {**drop** \| **set dscp dscp table** {*cir-markdown-map*}} [**violate** {**drop** \| **set dscp dscp table** {*pir-markdown-map*}}] | Polices **cir** in bits or as a percentage of the link rate. The **conform** action is taken if the data rate is **≤cir**. If **be** and **pir** are not specified, all other traffic takes the **violate** action. If **be** or **violate** are specified, the exceed action is taken if the data rate **≤ pir**. The actions are described in Information About Policing, on page 35. The data rates and link speeds are described in Police Command Data Rates, on page 40 and Police Command Burst Sizes, on page 40.<br>**Note** The default value of 200 ms is taken for bc and be, if bc and be are configured in ms/us. This limitation does not apply if bc and be are in bps. |
| **Step 5** | switch(config-pamp-c-qos)# **show policy-map** [**type qos**] [*policy-map-name*] | (Optional)<br>Displays information about all configured policy maps or a selected policy map of type QoS. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | switch# **show table-map** [**table-map-name**] | (Optional)<br>Displays information about the QoS table map. |
| **Step 7** | switch(config-pamp-c-qos)# **copy running-config startup-config** | (Optional)<br>Displays information about all configured policy maps or a selected policy map of type QoS. |

This example shows how to configure a 1-rate, 2-color policer that transmits if the data rate is within 200 milliseconds of traffic at 256000 bps and how to mark DSCP using the pir-markdown-map from the table map if the data rate is violated:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# policy-map pol1
switch(config-pmap-qos)# class class-default
switch(config-pmap-c-qos)# police cir 256000 bps conform transmit violate set dscp dscp
table pir-markdown-map
switch(config-pmap-c-qos)# show policy-map pol1


  Type qos policy-maps
  ====================

  policy-map type qos pol1
    class  class-default
      police cir 256000 bps bc 200 ms conform transmit violate set dscp dscp table
pir-markdown-map
switch(config-pmap-c-qos)# show table-map pir-markdown-map

  Table-map pir-markdown-map
    default copy
    from 10,12 to 14
    from 18,20 to 22
    from 26,28 to 30
    from 34,36 to 38

switch(config-pmap-c-qos)#
```

# Verifying the Policing Configuration

Use the following command to verify the configuration:

| Command | Description |
|---|---|
| **show policy-map** | Displays information about policy maps and policing. |

# Configuration Example for QoS Policing

This example shows a two- rate, three- color policer that sets the CoS to 4 if the data rate is within 300 kbps. It also shows how to mark down DSCP using the system-defined cir-markdown-map table map if the data

rate is within 750 kbps and how to mark down DSCP using the system-defined pir-markdown-map table map if the data rate is greater than 750 kbps:

```
switch(config)# policy-map ty qos 2rate3clr
switch(config-pmap-qos)# class  class1
switch(config-pmap-c-qos)# police cir 300 kbps pir 750 kbps conform
set-cos-transmit 4 exceed set dscp dscp table cir-markdown-map violate set dscp
dscp table pir-markdown-map
switch(config-pmap-c-qos)# show policy-map 2rate3clr
  Type qos policy-maps
  ====================

  policy-map type qos 2rate3clr
    class  class1
      police cir 300 kbps bc 200 ms pir 750 kbps be 200 ms conform set-cos-transmit 4
exceed set dscp dscp table cir-markdown-map violate set dscp dscp table pir-mar
kdown-map
```

This example shows a single-rate, two-color policer that transmits if the data rate is within 200 milliseconds of traffic at 600 kbps and drops packets otherwise:

```
switch(config)# policy-map ty qos 1rate2clr
switch(config-pmap-qos)# class  class2
switch(config-pmap-c-qos)# police cir 600 kbps conform transmit violate drop
switch(config-pmap-c-qos)# show policy-map 1rate2clr
  Type qos policy-maps
  ====================
  policy-map type qos 1rate2clr
    class  class2
      police cir 600 kbps bc 200 ms conform transmit violate drop
switch(config-pmap-c-qos)#
```

The following example shows how to configure a single-rate, three-color policer that polices traffic at 4,000,000 bits per second and allows normal or committed bursts of 200 kbytes and excess bursts of 400 kbytes. The policer transmits traffic that conforms to the policing rate, marks down the DSCP using the system-defined cir-markdown-map table map for traffic that exceeds the burst sizes, and drops traffic that violates the policing rate.

```
switch(config)# policy-map 1rate3clr
switch(config-pmap-qos)# class class1
switch(config-pmap-c-qos)# police cir 4 mbps bc 200 kbytes pir 4 mbps be 400 kbytes
conform transmit exceed set dscp dscp table cir-markdown-map violate drop
switch(config-pmap-c-qos)# show policy-map 1rate3clr

  Type qos policy-maps
  ====================

  policy-map type qos 1rate3clr
    class  class1
      police cir 4 mbps bc 200 kbytes pir 4 mbps be 400 kbytes conform transmit exceed set
 dscp dscp table cir-markdown-map violate drop
```

# Feature History for QoS Policing

This section provides the QoS policing release history.

| Feature Name | Release | Feature Information |
|---|---|---|
| QoS Policing | 5.2(1)SM1(5.1) | This feature was introduced |

CHAPTER 5

# Monitoring QoS Statistics

This chapter contains the following sections:


- Information About QoS Statistics, page 45
- Prerequisites for Monitoring QoS Statistics, page 45
- Enabling QoS Statistics, page 46
- Displaying QoS Statistics, page 47
- Clearing QoS Statistics, page 49
- Configuration Example for QoS Statistics, page 49
- Standards, page 50
- MIBs, page 50
- RFCs, page 50
- Technical Assistance, page 51
- Feature History for QoS Statistics, page 51


## Information About QoS Statistics

Statistics are maintained for each policy, class action, and match criteria per interface. You can enable or disable the collection of statistics globally using the **[no] qos statistics** command. You can display statistics using the **show policy-map interface** command, and you can clear statistics based on an interface or policy map with the **clear qos statistics** command. Statistics are enabled by default and can be disabled globally.

## Prerequisites for Monitoring QoS Statistics

You are logged in to the CLI in EXEC mode.

# Enabling QoS Statistics

**Note** By default, QoS statistics are enabled.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **qos statistics** | Enables QoS statistics on all interfaces. |
| **Step 3** | switch(config)# **show policy-map interface** | (Optional)<br>Displays the status of the global statistics and the configured policy maps on all interfaces. |
| **Step 4** | switch(config)# **copy running-config startup-config** | (Optional)<br>Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration. |

The following example shows how to enable QoS statistics:

```
switch# show policy-map interface
Global statistics status :   enabled

Vethernet1

  Service-policy (qos) input:   policy1
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      1 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      set dscp af31

    Class-map (qos):   class-default (match-any)
      1 packets

  Service-policy (qos) output:   policy1
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      9 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      set dscp af31

    Class-map (qos):   class-default (match-any)
      18645 packets

Vethernet2

  Service-policy (qos) output:   policy1
    policy statistics status:   enabled
```

```
    Class-map (qos):   class1 (match-all)
      18636 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      set dscp af31

    Class-map (qos):   class-default (match-any)
      19191 packets

Vethernet3

  Service-policy (qos) output:   policy1
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      9 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      set dscp af31

    Class-map (qos):   class-default (match-any)
      18644 packets

Vethernet4

  Service-policy (qos) input:   1r2c
    policy statistics status:   enabled

    Class-map (qos):   mat-dscp0 (match-all)
      568738560 packets
      Match: dscp 0
      police cir 90 mbps bc 200 ms
        conformed 37578738384 bytes, 0 bps action: set-dscp-transmit 10
        violated 0 bytes, 0 bps action: drop

  Service-policy (qos) output:   policy1
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      9 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      set dscp af31

    Class-map (qos):   class-default (match-any)
      18642 packets
```

# Displaying QoS Statistics

### Before You Begin

You know the interface for which statistics are needed.

**Note** Statistics for individual interfaces are often the most useful.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **show policy-map** [*policy-map-name* \| **interface** | Displays the specified statistics. |

| | Command or Action | Purpose |
|---|---|---|
| | [**brief** \| **ethernet** *interface_number* \| **output type qos** \| **port-channel** *number* \| **vethernet** *interface_number* \| **input type qos**] \| **type qos**] | To achieve the best result when your system has a large number of policies, use this command with specific arguments, such as specifying a particular interface or port channel.<br>• Use the **interface** keyword with the following keywords to display the service policy on an interface:<br>◦ **brief—**Displays a brief report of all policies attached to interfaces.<br>◦ **ethernet—**Displays statistics for an Ethernet interface.<br>◦ **input type qos—**Displays statistics for QoS input policies.<br>◦ **output type qos—**Displays statistics for QoS output policies.<br>◦ **port-channel—**Displays statistics for a port channel interface.<br>◦ **vethernet—**Displays the statistics for a vEthernet interface.<br>• Use the **type qos** keyword to display the type of policy map. |

This example shows you how to display a policy-map ethernet interface:

```
switch(config)# show policy-map interface port-channel 1
Global statistics status :   enabled

port-channel1

  Service-policy (qos) input:   2rate3clr
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      14 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      police cir 300 kbps bc 200 ms pir 750 kbps be 200 ms
        conformed 1260 bytes, 4384 bps action: set-cos-transmit 4
        exceeded 0 bytes, 0 bps action: set dscp dscp table cir-markdown-map
        violated 0 bytes, 0 bps action: set dscp dscp table pir-markdown-map

switch(config-if)# show policy-map interface vethernet 1

Global statistics status :   enabled

Vethernet1

  Service-policy (qos) input:   policy1
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      1 packets
      Match: access-group testacl1
      Match: packet length 3-6000
```

```
      set dscp af31

    Class-map (qos):   class-default (match-any)
      1 packets

  Service-policy (qos) output:   policy1
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      9 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      set dscp af31

    Class-map (qos):   class-default (match-any)
      19074 packets
```

# Clearing QoS Statistics

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch # **clear qos statistics** [**interface** {**ethernet** *interface_number* \| **port-channel** number \| **vethernet***interface_number* \| **output type qos** \| **input type qos**}] | Clears the specified QoS statistics. |

This example shows how to clear QoS statistics:

```
switch(config)# clear qos statistics interface port-channel 1
switch# show policy-map interface port-channel 1

Global statistics status :   enabled

port-channel1

  Service-policy (qos) input:   2rate3clr
    policy statistics status:   enabled

    Class-map (qos):   class1 (match-all)
      0 packets
      Match: access-group testacl1
      Match: packet length 3-6000
      police cir 300 kbps bc 200 ms pir 750 kbps be 200 ms
        conformed 0 bytes, 0 bps action: set-cos-transmit 4
        exceeded 0 bytes, 0 bps action: set dscp dscp table cir-markdown-map
        violated 0 bytes, 0 bps action: set dscp dscp table pir-markdown-map
switch#
```

# Configuration Example for QoS Statistics

This example shows how to display statistics for policy maps that are configured on interfaces:

```
switch(config)# show policy-map interface

Global statistics status :   enabled

Vethernet3

  Service-policy (qos) input:   new-policy
```

```
    policy statistics status:   enabled

    Class-map (qos):   class-default (match-any)
      59610700 packets
      set prec 5

Vethernet5

  Service-policy (qos) output:   new-policer
    policy statistics status:   enabled

    Class-map (qos):   new-class (match-all)
      344661013 packets
      Match: precedence 5
      police cir 900 mbps bc 200 ms
        conformed 505953339796 bytes, 899924196 bps action: transmit
        violated 12285218014 bytes, 22283000 bps action: drop
```

This example shows how to display the status of the global statistics and the configured policy maps on a specific interface:

```
switch(config)# show policy-map interface vethernet 3

Global statistics status :   enabled

Vethernet3

  Service-policy (qos) input:   policy-protoacl
    policy statistics status:   enabled

    Class-map (qos):   class-protoacl (match-any)
      132 packets
      Match: access-group protoacl
        132 packets
      set qos-group 100
```

# Standards

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-PROCESS-MIB | To locate and download MIBs, go to the following URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml |

# RFCs

No RFCs are supported by this feature.

# Technical Assistance

The Technical Assistance Center (TAC) home page contains 30,000 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/cisco/web/support/index.html

# Feature History for QoS Statistics

This section provides the QoS statistics release history.

| Feature Name | Release | Feature Information |
|---|---|---|
| QoS Statistics | 5.2(1)SM1(5.1) | This feature was introduced |