



## **Cisco Nexus 3000 NX-OS Quality of Service Configuration Guide, Release 5.0(3)U3(1)**

**First Published:** February 29, 2012

**Last Modified:** February 29, 2012

### **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

Text Part Number: OL-26545-01





## CONTENTS

---

### Preface

#### Preface v

Audience v

Document Conventions v

Related Documentation for Nexus 3000 Series NX-OS Software vi

Obtaining Documentation and Submitting a Service Request viii

---

### CHAPTER 1

#### New and Changed Information for this Release 1

New and Changed Information 1

---

### CHAPTER 2

#### Overview 3

Quality of Service Overview 3

---

### CHAPTER 3

#### Configuring QoS 5

Quality of Service 5

Modular QoS CLI 5

System Classes 6

Default System Classes 7

Information About Policy Types 7

Trust Boundaries 9

Ingress Classification Policies 10

Egress Queuing Policies 10

QoS for Traffic Directed to the CPU 10

QoS Configuration Guidelines and Limitations 11

Configuring System Classes 11

Configuring Class Maps 11

Configuring ACL Classification 12

Configuring CoS Classification 13

Configuring DSCP Classification	14
Configuring IP RTP Classification	15
Configuring Precedence Classification	16
Creating Policy Maps	17
Configuring Type QoS Policies	19
Configuring Type Network QoS Policies	19
Configuring Type Queuing Policies	20
Information About Marking	21
Configuring CoS Marking	22
Configuring DSCP Marking	22
Configuring IP Precedence Marking	24
QoS Configurations for Layer 3 Routing	25
Required CoS Marking Configuration in a Layer 3 Topology	25
Configuring Layer 3 Multicast Queuing	27
Configuring a Service Policy for a Layer 3 Interface	27
Changing the Bandwidth Allocated to Unicast and Multicast Traffic	28
Attaching the System Service Policy	29
Restoring the Default System Service Policies	29
Enabling the Jumbo MTU	30
Verifying the Jumbo MTU	30
Configuring QoS on Interfaces	31
Configuring Untagged CoS	31
Configuring Interface Service Policy	32
Verifying QoS Configuration	33
Monitoring the QoS Packet Buffer	39



## Preface

---

This preface contains the following sections:

- [Audience, page v](#)
- [Document Conventions, page v](#)
- [Related Documentation for Nexus 3000 Series NX-OS Software, page vi](#)
- [Obtaining Documentation and Submitting a Service Request, page viii](#)

## Audience

This publication is for experienced network administrators who configure and maintain Cisco Nexus Series devices.

## Document Conventions

Command descriptions use the following conventions:

Convention	Description
<b>bold</b>	Bold text indicates the commands and keywords that you enter literally as shown.
<i>Italic</i>	Italic text indicates arguments for which the user supplies the values.
[x]	Square brackets enclose an optional element(keyword or argument).
[x   y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x   y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.

Convention	Description
[x {y   z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
<i>variable</i>	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.

Examples use the following conventions:

Convention	Description
<code>screen font</code>	Terminal sessions and information the switch displays are in screen font.
<b><code>boldface screen font</code></b>	Information you must enter is in boldface screen font.
<i><code>italic screen font</code></i>	Arguments for which you supply values are in italic screen font.
<>	Nonprinting characters, such as passwords, are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

This document uses the following conventions:



**Note**

Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.



**Caution**

Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

## Related Documentation for Nexus 3000 Series NX-OS Software

The entire Cisco NX-OS 3000 Series documentation set is available at the following URL:

[http://www.cisco.com/en/US/products/ps11541/tsd\\_products\\_support\\_series\\_home.html](http://www.cisco.com/en/US/products/ps11541/tsd_products_support_series_home.html)

## Release Notes

The release notes are available at the following URL:

[http://www.cisco.com/en/US/products/ps11541/prod\\_release\\_notes\\_list.html](http://www.cisco.com/en/US/products/ps11541/prod_release_notes_list.html)

## Installation and Upgrade Guides

The installation and upgrade guides are available at the following URL:

[http://www.cisco.com/en/US/products/ps11541/prod\\_installation\\_guides\\_list.html](http://www.cisco.com/en/US/products/ps11541/prod_installation_guides_list.html)

The documents in this category include:

- *Cisco Nexus 5000 Series, Cisco Nexus 3000 Series, and Cisco Nexus 2000 Series Safety Information and Documentation*
- *Regulatory, Compliance, and Safety Information for the Cisco Nexus 5000 Series, Cisco Nexus 3000 Series, and Cisco Nexus 2000 Series*
- *Cisco Nexus 3000 Series Hardware Installation Guide*

## Configuration Guides

The configuration guides are available at the following URL:

[http://www.cisco.com/en/US/products/ps11541/products\\_installation\\_and\\_configuration\\_guides\\_list.html](http://www.cisco.com/en/US/products/ps11541/products_installation_and_configuration_guides_list.html)

The documents in this category include:

- *Fundamentals Configuration Guide*
- *Interfaces Configuration Guide*
- *Layer 2 Switching Configuration Guide*
- *Multicast Configuration Guide*
- *Quality of Service Configuration Guide*
- *Security Configuration Guide*
- *System Management Configuration Guide*
- *Unicast Routing Configuration Guide*
- *Verified Scalability Guide for Cisco NX-OS*

## Technical References

The technical references are available at the following URL:

[http://www.cisco.com/en/US/products/ps11541/prod\\_technical\\_reference\\_list.html](http://www.cisco.com/en/US/products/ps11541/prod_technical_reference_list.html)

## Error and System Messages

The error and system message reference guides are available at the following URL:

[http://www.cisco.com/en/US/products/ps11541/products\\_system\\_message\\_guides\\_list.html](http://www.cisco.com/en/US/products/ps11541/products_system_message_guides_list.html)

# Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.





# CHAPTER 1

## New and Changed Information for this Release

The following table provides an overview of the significant changes to this guide for this current release. The table does not provide an exhaustive list of all changes made to the configuration guides or of the new features in this release.

- [New and Changed Information, page 1](#)

## New and Changed Information

The latest version of this document is available at the following Cisco website:

[http://www.cisco.com/en/US/products/ps11541/products\\_installation\\_and\\_configuration\\_guides\\_list.html](http://www.cisco.com/en/US/products/ps11541/products_installation_and_configuration_guides_list.html).

To check for the latest information about Cisco NX-OS for the Cisco Nexus 3000 Series switch, see the *Cisco Nexus 3000 Series Release Notes* available at the following Cisco website:

[http://www.cisco.com/en/US/products/ps11541/prod\\_release\\_notes\\_list.html](http://www.cisco.com/en/US/products/ps11541/prod_release_notes_list.html)

This table summarizes the new and changed features documented in the *Cisco Nexus 3000 Series NX-OS Quality of Service Configuration Guide, Release 5.0(3)U3(1)* and tells you where they are documented.

**Table 1: New and Changed Quality of Service Features for Cisco NX-OS Release 5.0(3)U3(1)**

Feature	Description	Added or Changed in Release	Where Documented
IPv6	Added information about IPv6 support for configuring QoS.	5.0(3)U3(1)	Configuring QoS





## CHAPTER 2

# Overview

---

This chapter contains the following sections:

- [Quality of Service Overview, page 3](#)

## Quality of Service Overview

This document describes the configurable Cisco NX-OS Quality of Service (QoS) features. You use the QoS features to provide the most desirable flow of traffic through a network. QoS allows you to classify the network traffic, prioritize the traffic flow, and provide congestion avoidance. The control of traffic is based on the fields in the packets that flow through the system. You use the Modular QoS CLI (MQC) to create the traffic classes and policies of the QoS features.

QoS features are applied using QoS policies and queuing policies, as follows:

- QoS policies include classification and marking features.
- Queuing policies use the queuing and scheduling features.
- Network QoS policies include configuring MTU.





## CHAPTER 3

# Configuring QoS

---

This chapter describes how to configure quality of service (QoS) on Cisco Nexus 3000 Series devices. It contains the following sections:

- [Quality of Service, page 5](#)
- [QoS Configuration Guidelines and Limitations, page 11](#)
- [Configuring System Classes, page 11](#)
- [Configuring QoS on Interfaces, page 31](#)
- [Verifying QoS Configuration, page 33](#)
- [Monitoring the QoS Packet Buffer, page 39](#)

## Quality of Service

The configurable Cisco NX-OS quality of service (QoS) features allow you to classify the network traffic, prioritize the traffic flow, and provide congestion avoidance.

The default QoS configuration on the device provides best-effort service for Ethernet traffic. QoS can be configured to provide additional classes of service for Ethernet traffic. Cisco NX-OS QoS features are configured using Cisco Modular QoS CLI (MQC).

**Note**

---

In the event of congestion or collisions, Ethernet will drop packets. The higher level protocols detect the missing data and retransmit the dropped packets.

---

## Modular QoS CLI

The Cisco Modular QoS CLI (MQC) provides a standard set of commands for configuring QoS.

You can use MQC to define additional traffic classes and to configure QoS policies for the whole system and for individual interfaces. Configuring a QoS policy with MQC consists of the following steps:

- 1 Define traffic classes.

- 2 Associate policies and actions with each traffic class.
- 3 Attach policies to logical or physical interfaces as well as at the global system level.

MQC provides two command types to define traffic classes and policies:

#### **class-map**

Defines a class map that represents a class of traffic based on packet-matching criteria. Class maps are referenced in policy maps.

The class map classifies incoming packets based on matching criteria, such as the IEEE 802.1p CoS value. Unicast and multicast packets are classified.

#### **policy-map**

—Defines a policy map that represents a set of policies to be applied on a class-by-class basis to class maps.

The policy map defines a set of actions to take on the associated traffic class, such as limiting the bandwidth or dropping packets.

You define the following class-map and policy-map object types when you create them:

#### **network-qos**

Defines MQC objects that you can use for system level related actions.

#### **qos**

Defines MQC objects that you can use for classification.

#### **queuing**

Defines MQC objects that you can use for queuing and scheduling.




---

**Note** The qos type is the default for the **class-map** and **policy-map** commands, but not for the **service-policy** which requires that you specify an explicit type.

---

You can attach policies to interfaces or EtherChannels as well as at the global system level by using the **service-policy** command.

You can view all or individual values for MQC objects by using the **show class-map** and **show policy-map** commands.

An MQC target is an entity (such as an Ethernet interface) that represents a flow of packets. A service policy associates a policy map with an MQC target and specifies whether to apply the policy on incoming or outgoing packets. This mapping enables the configuration of QoS policies such as marking, bandwidth allocation, buffer allocation, and so on.

## System Classes

The system qos is a type of MQC target. You use a service-policy to associate a policy map with the system qos target. A system qos policy applies to all interfaces on the switch unless a specific interface has an overriding service-policy configuration. The system qos policies are used to define system classes, the classes

of traffic across the entire switch, and their attributes. To ensure QoS consistency (and for ease of configuration), the switch distributes the system class parameter values to all its attached network adapters using the Data Center Bridging Exchange (DCBX) protocol.

If service policies are configured at the interface level, the interface-level policy always takes precedence over system class configuration or defaults.

On the Cisco Nexus 3000 Series switch, a system class is uniquely identified by a qos-group value. A total of eight system classes are supported. The Cisco Nexus 3000 Series switch supports one default class which is always present on the switch. Up to seven additional system classes can be created by the administrator.

## Default System Classes

The switch provides the following system classes:

- Drop system class

By default, the software classifies all unicast and multicast Ethernet traffic into the default drop system class. This class is identified by qos-group 0.

This class is created automatically when the system starts up (the class is named **class-default** in the CLI). You cannot delete this class and you cannot change the match criteria associated with the default class.

## Information About Policy Types

The switch supports a number of policy types. You create class maps in the policy types.

There are three policy types. The following QoS parameters can be specified for each type of class:

- Type network-qos—A network-qos policy is used to instantiate system classes and associate parameters with those classes that are of system-wide scope.
  - Classification—The traffic that matches this class are as follows:
    - QoS Group—A class-map of type network-qos identifies a system-class and is matched by its associated qos-group.
  - Policy—The actions that are performed on the matching traffic are as follows:



---

**Note** A network-qos policy can only be attached to the system qos target.

---

- MTU—The MTU that needs to be enforced for the traffic that is mapped to a system class.



---

**Note** Cisco Nexus 3000 Series supports one MTU for all classes for all ports.

---

- Set CoS value—This configuration is used to mark 802.1p values for all traffic mapped to this system class.

- Congestion Control WRED—WRED anticipates and avoids congestion before congestion occurs. WRED drops packets, based on the average queue length exceeding a specific threshold value, to indicate congestion. You can configure congestion avoidance with WRED in egress policy maps. By default, tail-drop is the congestion control mechanism. To enable WRED, use the `congestion-control random-detect` command in `network-qos` policy map mode.
- ECN—ECN is an extension to WRED that marks packets instead of dropping them when the average queue length exceeds a specific threshold value. When configured with the WRED Explicit Congestion Notification feature, routers and end hosts use this marking as a signal that the network is congested to slow down sending packets. To enable ECN use the **congestion-control random-detect ecn** command in the `network-qos` policy map mode.




---

**Note** Enabling WRED/ECN on a class on a `network-qos` policy implies that WRED/ECN is enabled for all ports in the system.

---

- Type queuing—A type queuing policy is used to define the scheduling characteristics of the queues associated with system classes.

Cisco Nexus 3000 Series supports type queuing in the egress direction.




---

**Note** Some configuration parameters when applied to an EtherChannel are not reflected on the configuration of the member ports.

---

- Classification—The traffic that matches this class are as follows:
  - QoS Group—A class-map of type queuing identifies a system-class and is matched by its associated `qos-group`.
- Policy—The actions that are performed on the matching traffic are as follows:




---

**Note** These policies can be attached to the system `qos` target or to any interface. The output queuing policy is used to configure output queues on the switch associated with system classes.

---

- Bandwidth—Sets the guaranteed scheduling deficit weighted round robin (DWRR) percentage for the system class.
  - Priority—Sets a system class for strict-priority scheduling. Only one system class can be configured for priority in a given queuing policy.
- Type `qos`—A type `qos` policy is used to classify traffic that is based on various Layer 2, Layer 3, and Layer 4 fields in the frame and to map it to system classes.





---

**Note** Some configuration parameters when applied to an EtherChannel are not reflected on the configuration of the member ports.

---

- Classification—The traffic that matches this class are as follows:
  - Access Control Lists—Classifies traffic based on the criteria in existing ACLs.
  - Class of Service—Matches traffic based on the CoS field in the frame header.
  - DSCP—Classifies traffic based on the Differentiated Services Code Point (DSCP) value in the DiffServ field of the IP header.
  - IP Real Time Protocol—Classifies traffic on the port numbers used by real-time applications.
  - Precedence—Classifies traffic based on the precedence value in the type of service (ToS) field of the IP header.
- Policy—The actions that are performed on the matching traffic are as follows:



---

**Note** This policy can be attached to the system or to any interface. It applies to input traffic only.

---

- QoS Group—Sets the qos-group corresponding to the system class this traffic flow is mapped to.
  - Cisco Nexus 3000 Series supports the following:
    - 8 qos-groups
    - 8 queues for unicast
    - 4 queues for multicast

By default, 2 qos-groups each are mapped to 1 multicast queue. The mapping is qos-group 0 and 1 are mapped to a multicast queue, qos-group 2 and 3 are mapped to the next and so forth.

## Trust Boundaries

The trust boundary is enforced by the incoming interface as follows:

- By default, all Ethernet interfaces are trusted interfaces. The 802.1p CoS and DSCP are preserved unless the marking is configured. There is no default CoS to queue and DSCP to queue mapping. You can define and apply a policy to create these mappings. By default, without a user defined policy, all traffic is assigned to the default queue.

- Any packet that is not tagged with an 802.1p CoS value is classified into the default drop system class. If the untagged packet is sent over a trunk, it is tagged with the default untagged CoS value, which is zero.
- You can override the default untagged CoS value for an Ethernet interface or port channel.
- You can override the default untagged CoS value for an Ethernet interface or a port channel interface using the **untagged cos** *cos-value* command.
- You can override the default untagged Cos value for an Ethernet or a Layer 3 interface or a port channel interface using the **untagged cos** *cos-value* command.

After the system applies the untagged CoS value, QoS functions the same as for a packet that entered the system tagged with the CoS value.

## Ingress Classification Policies

You use classification to partition traffic into classes. You classify the traffic based on the port characteristics (CoS field) or the packet header fields that include IP precedence, Differentiated Services Code Point (DSCP), and Layer 2 to Layer 4 parameters. The values used to classify traffic are called match criteria. When you define a traffic class, you can specify multiple match criteria or you can determine the traffic class by matching any or all criteria.

Traffic that fails to match any class is assigned to a default class of traffic called class-default.

## Egress Queuing Policies

You can associate an egress policy map with an Ethernet interface to guarantee the bandwidth for the specified traffic class or to configure the egress queues.

The bandwidth allocation limit applies to all traffic on the interface.

Each Ethernet interface supports up to six queues, one for each system class. The queues have the following default configuration:

- In addition to the six queues, control traffic that is destined for the CPU uses strict priority queues. These queues are not accessible for user configuration.
- Standard Ethernet traffic in the default drop system class is assigned a queue. This queue uses WRR scheduling with 50 percent of the bandwidth.

If you add a system class, a queue is assigned to the class. You must reconfigure the bandwidth allocation on all affected interfaces. Bandwidth is not dedicated automatically to user-defined system classes.

You can configure a strict priority queue. This queue is serviced before all other queues except the control traffic queue (which carries control rather than data traffic).

## QoS for Traffic Directed to the CPU

The device automatically applies QoS policies to traffic that is directed to the CPU to ensure that the CPU is not flooded with packets. Control traffic, such as bridge protocol data units (BPDU) frames, is given higher priority to ensure delivery.

# QoS Configuration Guidelines and Limitations

To maintain optimal switch performance, follow these guidelines when configuring system classes and policies:

- Switch resources (such as buffers, virtual output queues, and egress queues) are partitioned based on the default and user-defined system classes. Cisco NX-OS automatically adjusts the resource allocation to accommodate the configured system classes.
- WRED and ECN configuration are supported only on unicast flows. WRED and ECN configuration do not affect other flows such as multicast, broadcast, and unknown unicast.
- WRED and ECN configuration is not supported on a class mapped to qos-group 1.

When configuring EtherChannels, note the following guidelines:

- The service policy configured on an EtherChannel applies to all member interfaces.

## Configuring System Classes

### Configuring Class Maps

You can create or modify a class map with the **class-map** command. The class map is a named object that represents a class of traffic. In the class map, you specify a set of match criteria for classifying the packets. You can then reference class maps in policy maps.



#### Note

The class map type default is type qos and its match criteria default is match-all.

#### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map</b> [ <b>type {network-qos   qos  </b> <b>queuing}</b> ] <i>class-map name</i>	Creates or accesses a named object that represents the specified class of traffic.  Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.  The three class-map configuration modes are as follows: <ul style="list-style-type: none"> <li>• network-qos—Network-wide (global) mode. CLI prompt: switch(config-cmap-nq)#</li> <li>• qos—Classification mode; this is the default mode. CLI prompt: switch(config-cmap-qos)#</li> <li>• queuing—Queuing mode. CLI prompt: switch(config-cmap-que)#</li> </ul>

	Command or Action	Purpose
<b>Step 3</b>	switch(config)# <b>class-map</b> [ <b>type qos</b> ] [ <b>match-all</b>   <b>match-any</b> ] <i>class-map name</i>	(Optional) Specifies that packets must match any or all criteria that is defined for a class map. <ul style="list-style-type: none"> <li>• <b>match-all</b>—Classifies traffic if packets match all criteria that is defined for a specified class map (for example, if both the defined CoS and the ACL criteria match).</li> <li>• <b>match-any</b>—Classifies traffic if packets match any criteria that is defined for a specified class map (for example, if either the CoS or the ACL criteria matches).</li> </ul> Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 4</b>	switch(config)# <b>no class-map</b> [ <b>type {network-qos   qos  </b> <b>queuing}</b> ] <i>class-name</i>	(Optional) Deletes the specified class map. <b>Note</b> You cannot delete the system-defined class map: <b>class-default</b> . Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.

## Configuring ACL Classification

You can classify traffic by matching packets based on an existing access control list (ACL). Traffic is classified by the criteria defined in the ACL. The **permit** and **deny** ACL keywords are ignored in the matching; even if a match criteria in the access-list has a **deny** action, it is still used for matching for this class.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map type qos</b> <i>class-name</i>	Creates a named object that represents a class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-cmap-qos)# <b>match</b> <b>access-group name</b> <i>acl-name</i>	Configures a traffic class by matching packets based on the <i>acl-name</i> . The <b>permit</b> and <b>deny</b> ACL keywords are ignored in the matching. <b>Note</b> You can only define a single ACL in a class map. You cannot add any other match criteria to a class with a <b>match access-group</b> defined.

	Command or Action	Purpose
<b>Step 4</b>	switch(config-cmap-qos)# <b>no match access-group name <i>acl-name</i></b>	(Optional) Removes the match from the traffic class.

This example shows how to classify traffic by matching packets based on existing ACLs:

```
switch# configure terminal
switch(config)# class-map type qos class_acl
switch(config-cmap-qos)# match access-group name acl-01
```

Use the **show class-map** command to display the ACL class-map configuration:

```
switch# show class-map class_acl
```

## Configuring CoS Classification

You can classify traffic based on the class of service (CoS) in the IEEE 802.1Q header. This 3-bit field is defined in IEEE 802.1p to support QoS traffic classes. CoS is encoded in the high order 3 bits of the VLAN ID Tag field and is referred to as *user\_priority*.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map type qos <i>class-name</i></b>	Creates a named object that represents a class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-cmap-qos)# <b>match cos <i>cos-value</i></b>	Specifies the CoS value to match for classifying packets into this class. You can configure a CoS value in the range of 0 to 7.
<b>Step 4</b>	switch(config-cmap-qos)# <b>no match cos <i>cos-value</i></b>	(Optional) Removes the match from the traffic class.

This example shows how to classify traffic by matching packets based on a defined CoS value:

```
switch# configure terminal
switch(config)# class-map type qos match-any class_cos
switch(config-cmap-qos)# match cos 4, 5-6
```

Use the **show class-map** command to display the CoS value class-map configuration:

```
switch# show class-map class_cos
```

## Configuring DSCP Classification

You can classify traffic based on the Differentiated Services Code Point (DSCP) value in the DiffServ field of the IP header (either IPv4 or IPv6).

**Table 2: Standard DSCP Values**

Value	List of DSCP Values
af11	AF11 dscp (001010)—decimal value 10
af12	AF12 dscp (001100)—decimal value 12
af13	AF13 dscp (001110)—decimal value 14
af21	AF21 dscp (010010)—decimal value 18
af22	AF22 dscp (010100)—decimal value 20
af23	AF23 dscp (010110)—decimal value 22
af31	AF31 dscp (011010)—decimal value 26
af32	AF32 dscp (011100)—decimal value 28
af33	AF33 dscp (011110)—decimal value 30
af41	AF41 dscp (100010)—decimal value 34
af42	AF42 dscp (100100)—decimal value 36
af43	AF43 dscp (100110)—decimal value 38
cs1	CS1 (precedence 1) dscp (001000)—decimal value 8
cs2	CS2 (precedence 2) dscp (010000)—decimal value 16
cs3	CS3 (precedence 3) dscp (011000)—decimal value 24
cs4	CS4 (precedence 4) dscp (100000)—decimal value 32
cs5	CS5 (precedence 5) dscp (101000)—decimal value 40
cs6	CS6 (precedence 6) dscp (110000)—decimal value 48

Value	List of DSCP Values
cs7	CS7 (precedence 7) dscp (111000)—decimal value 56
default	Default dscp (000000)—decimal value 0
ef	EF dscp (101110)—decimal value 46

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map type qos</b> <i>class-name</i>	Creates a named object that represents a class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-cmap-qos)# <b>match</b> <b>dscp</b> <i>dscp-list</i>	Configures the traffic class by matching packets based on the values in the <i>dscp-list</i> variable. For a list of DSCP values, see the Standard DSCP Values table.
<b>Step 4</b>	switch(config-cmap-qos)# <b>no match</b> <b>dscp</b> <i>dscp-list</i>	(Optional) Removes the match from the traffic class. For a list of DSCP values, see the Standard DSCP Values table.

This example shows how to classify traffic by matching packets based on the DSCP value in the DiffServ field of the IP header:

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

Use the **show class-map** command to display the DSCP class-map configuration:

```
switch# show class-map class_dscp
```

## 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 classify based on UDP port ranges, which are likely to target applications using RTP.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map type qos</b> <i>class-name</i>	Creates a named object that represents a class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-cmap-qos)# <b>match ip</b> <b>rtp</b> <i>port-number</i>	Configures the traffic class by matching packets based on a range of lower and upper UDP port numbers, which is likely to target applications using RTP. Values can range from 2000 to 65535.
<b>Step 4</b>	switch(config-cmap-qos)# <b>no match</b> <b>ip rtp</b> <i>port-number</i>	(Optional) Removes the match from the traffic class.

This example shows how to classify traffic by matching packets based on UDP port ranges that are typically used by RTP applications:

```
switch# configure terminal
switch(config)# class-map type qos match-any class_rtp
switch(config-cmap-qos)# match ip rtp 2000-2100, 4000-4100
```

Use the **show class-map** command to display the RTP class-map configuration:

```
switch# show class-map class_rtp
```

## Configuring Precedence Classification

You can classify traffic based on the precedence value in the type of service (ToS) byte field of the IP header (either IPv4 or IPv6). The following table shows the precedence values:

**Table 3: Precedence Values**

Value	List of Precedence Values
<0-7>	IP precedence value
critical	Critical precedence (5)
flash	Flash precedence (3)
flash-override	Flash override precedence (4)
immediate	Immediate precedence (2)
internet	Internetwork control precedence (6)



Value	List of Precedence Values
network	Network control precedence (7)
priority	Priority precedence (1)
routine	Routine precedence (0)

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map type qos match-any class-name</b>	Creates a named object that represents a class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-cmap-qos)# <b>match precedence precedence-values</b>	Configures the traffic class by matching packets based on precedence values. For a list of precedence values, see the Precedence Values table.
<b>Step 4</b>	switch((config-cmap-qos)# <b>no match precedence precedence-values</b>	(Optional) Removes the match from the traffic class. For a list of precedence values, see the Precedence Values table.

This example shows how to classify traffic by matching packets based on the precedence value in the ToS byte field of the IP header:

```
switch# configure terminal
switch(config)# class-map type qos match-any class_precedence
switch(config-cmap-qos)# match precedence 1-2, critical
```

Use the **show class-map** command to display the IP precedence value class-map configuration:

```
switch# show class-map class_precedence
```

## Creating Policy Maps

The **policy-map** command is used to create a named object that represents a set of policies that are to be applied to a set of traffic classes.

The switch provides one default system class: a drop class for best-effort service (class-default). You can define up to four additional system classes for Ethernet traffic.

The following predefined policy maps are used as default service policies:

- network-qos: default-nq-policy
- Input qos: default-in-policy
- Output queuing: default-out-policy

You need to create a policy map to specify the policies for any user-defined class. In the policy map, you can configure the QoS parameters for each class. You can use the same policy map to modify the configuration of the default classes.

The switch distributes all the policy-map configuration values to the attached network adapters.

### Before You Begin

Before creating the policy map, define a class map for each new system class.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>policy-map</b> [type { <b>network-qos</b>   <b>qos</b>   <b>queuing</b> }] <i>policy-name</i>	Creates a named object representing a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.  The three policy-map configuration modes are as follows: <ul style="list-style-type: none"> <li>• <b>network-qos</b>—Network-wide (global) mode. CLI prompt: switch(config-pmap-nq)#</li> <li>• <b>qos</b>—Classification mode; this is the default mode. CLI prompt: switch(config-pmap-qos)#</li> <li>• <b>queuing</b>—Queuing mode. CLI prompt: switch(config-pmap-que)#</li> </ul>
<b>Step 3</b>	switch(config)# <b>no policy-map</b> [type { <b>network-qos</b>   <b>qos</b>   <b>queuing</b> }] <i>policy-name</i>	(Optional) Deletes the specified policy map.
<b>Step 4</b>	switch(config-pmap)# <b>class</b> [type { <b>network-qos</b>   <b>qos</b>   <b>queuing</b> }] <i>class-name</i>	Associates a class map with the policy map, and enters configuration mode for the specified system class. The three class-map configuration modes are as follows: <ul style="list-style-type: none"> <li>• <b>network-qos</b>—Network-wide (global) mode. CLI prompt: switch(config-pmap-c-nq)#</li> <li>• <b>qos</b>—Classification mode; this is the default mode. CLI prompt: switch(config-pmap-c-qos)#</li> <li>• <b>queuing</b>—Queuing mode. CLI prompt: switch(config-pmap-c-que)#</li> </ul> <p><b>Note</b> The associated class map must be the same type as the policy-map type.</p>
<b>Step 5</b>	switch(config-pmap)# <b>no class</b> [type { <b>network-qos</b>   <b>qos</b>   <b>queuing</b> }] <i>class-name</i>	(Optional) Deletes the class map association.

## Configuring Type QoS Policies

Type qos policies are used for classifying the traffic of a specific system class identified by a unique qos-group value. A type qos policy can be attached to the system or to individual interfaces for ingress traffic only.

You can set a maximum of five qos groups for ingress traffic.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>policy-map type qos</b> <i>policy-name</i>	Creates a named object that represents a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-pmap-qos)# <b>class type qos</b> <i>class-name</i>	Associates a class map with the policy map, and enters configuration mode for the specified system class. <b>Note</b> The associated class map must be the same type as the policy map type.
<b>Step 4</b>	switch(config-pmap-c-qos)# <b>set qos-group</b> <i>qos-group-value</i>	Configures one or more <b>qos-group</b> values to match on for classification of traffic into this class map. The list below identifies the ranges of the <i>qos-group-value</i> . There is no default value. <b>Note</b> The switch can only support a maximum of five qos-groups within this range.
<b>Step 5</b>	switch(config-pmap-c-qos)# <b>no set qos-group</b> <i>qos-group-value</i>	(Optional) Removes the <b>qos-group</b> values from this class.

This example shows how to define a type qos policy map:

```
switch# configure terminal
switch(config)# policy-map type qos policy-s1
switch(config-pmap-qos)# class type qos class-s1
switch(config-pmap-c-qos)# set qos-group 2
```

## Configuring Type Network QoS Policies

Type network qos policies can only be configured on the system qos attachment point. They are applied to the entire switch for a particular class.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.

	Command or Action	Purpose
<b>Step 2</b>	<code>switch(config)# <b>policy-map type network-qos</b> <i>policy-name</i></code>	Creates a named object that represents a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	<code>switch(config-pmap-nq)# <b>class type network-qos</b> <i>class-name</i></code>	Associates a class map with the policy map, and enters configuration mode for the specified system class. <b>Note</b> The associated class map must be the same type as the policy map type.
<b>Step 4</b>	<code>switch(config-pmap-c-nq)# <b>mtu</b> <i>mtu-value</i></code>	Specifies the MTU value in bytes. <b>Note</b> The <i>mtu-value</i> that you configure must be less than the value set by the <b>system jumbomtu</b> command.
<b>Step 5</b>	<code>switch(config-pmap-c-nq)# <b>no mtu</b></code>	(Optional) Resets the MTU value in this class.
<b>Step 6</b>	<code>switch(config-pmap-c-nq)# <b>congestion-control random-detect</b></code>	(Optional) Configure congestion avoidance with WRED in egress policy maps. By default, tail-drop is the congestion control mechanism..
<b>Step 7</b>	<code>switch(config-pmap-c-nq)# <b>congestion-control random-detect ecn</b></code>	(Optional) Marks packets instead of dropping them when the average queue length exceeds a specific threshold value. Routers and end hosts use this marking as a signal that the network is congested to slow down sending packets.
<b>Step 8</b>	<code>switch(config-pmap-c-nq)# <b>set cos</b> <i>cos-value</i></code>	Specifies a 802.1Q CoS value which is used to mark packets on this interface. The value range is from 0 to 7.
<b>Step 9</b>	<code>switch(config-pmap-c-nq)# <b>no set cos</b> <i>cos-value</i></code>	(Optional) Disables the marking operation in this class.

This example shows how to define a type network-qos policy map:

```
switch# configure terminal
switch(config)# policy-map type network-qos policy-que1
switch(config-pmap-nq)# class type network-qos class-que1
switch(config-pmap-c-nq)# mtu 5000
switch(config-pmap-c-nq)# set cos 4
```

## Configuring Type Queuing Policies

Type queuing policies are used for scheduling and buffering the traffic of a specific system class. A type queuing policy is identified by its qos-group and can be attached to the system or to individual interfaces for input or output traffic.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>policy-map type queuing</b> <i>policy-name</i>	Creates a named object that represents a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-pmap-que)# <b>class type queuing</b> <i>class-name</i>	Associates a class map with the policy map, and enters configuration mode for the specified system class.
<b>Step 4</b>	switch(config-pmap-c-que)# <b>bandwidth percent</b> <i>percentage</i>	Specifies the guaranteed percentage of interface bandwidth allocated to this class. By default, no bandwidth is specified for a class.  <b>Note</b> Before you can successfully allocate bandwidth to the class, you must first reduce the default bandwidth configuration on class-default and class-feoe.
<b>Step 5</b>	switch(config-pmap-c-que)# <b>no bandwidth percent</b> <i>percentage</i>	(Optional) Removes the bandwidth specification from this class.
<b>Step 6</b>	switch(config-pmap-c-que)# <b>priority</b>	Specifies that traffic in this class is mapped to a strict priority queue.  <b>Note</b> Only one class in each policy map can have strict priority set on it.
<b>Step 7</b>	switch(config-pmap-c-que)# <b>no priority</b>	(Optional) Removes the strict priority queuing from the traffic in this class.

This example shows how to define a type queuing policy map:

```
switch# configure terminal
switch(config)# policy-map type queuing policy-queue1
switch(config-pmap-que)# class type queuing class-queue1
switch(config-pmap-c-que)# bandwidth 20
```

## Information About Marking

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, IP precedence, and Differentiated Service Code Point (DSCP).

You can use marking commands in traffic classes that are referenced in a policy map. The marking features that you can configure are listed below:

- DSCP

- IP precedence
- CoS

## Configuring CoS Marking

The value of the CoS field is recorded in the high-order three bits of the VLAN ID Tag field in the IEEE 802.1Q header.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config) # <b>policy-map [type network-qos] policy-map name</b>	Creates or accesses the policy map named <i>policy-map-name</i> and enters policy-map mode.  The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
<b>Step 3</b>	switch(config-pmap-nq) # <b>class [type network-qos] {class-map name  class-default}</b>	Creates a reference to <i>class-map-name</i> and enters policy-map class configuration mode.  Use the <b>class-default</b> keyword to select all traffic that is not currently matched by classes in the policy map.
<b>Step 4</b>	switch(config-pmap-c-nq) # <b>set cos cos-value</b>	Specifies the CoS value to <i>cos-value</i> .  The <i>cos-value</i> can range from 0 to 7.  <b>Note</b> This command is only supported for egress policies.

## Configuring DSCP Marking

You can set the DSCP value in the six most significant bits of the DiffServ field of the IP header to a specified value. You can enter numeric values from 0 to 60, in addition to the standard DSCP values shown in the table below:



### Note

You can set DSCP or IP Precedence but you can not set both values because they modify the same field in the IP packet.

**Table 4: Standard DSCP Values**

<b>Value</b>	<b>List of DSCP Values</b>
af11	AF11 dscp (001010)—decimal value 10
af12	AF12 dscp (001100)—decimal value 12
af13	AF13 dscp (001110)—decimal value 14
af21	AF21 dscp (010010)—decimal value 18
af22	AF22 dscp (010100)—decimal value 20
af23	AF23 dscp (010110)—decimal value 22
af31	AF31 dscp (011010)—decimal value 26
af32	AF40 dscp (011100)—decimal value 28
af33	AF33 dscp (011110)—decimal value 30
af41	AF41 dscp (100010)—decimal value 34
af42	AF42 dscp (100100)—decimal value 36
af43	AF43 dscp (100110)—decimal value 38
cs1	CS1 (precedence 1) dscp (001000)—decimal value 8
cs2	CS2 (precedence 2) dscp (010000)—decimal value 16
cs3	CS3 (precedence 3) dscp (011000)—decimal value 24
cs4	CS4 (precedence 4) dscp (100000)—decimal value 32
cs5	CS5 (precedence 5) dscp (101000)—decimal value 40
cs6	CS6 (precedence 6) dscp (110000)—decimal value 48
cs7	CS7 (precedence 7) dscp (111000)—decimal value 56
default	Default dscp (000000)—decimal value 0

Value	List of DSCP Values
ef	EF dscp (101110)—decimal value 46

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<code>confi t</code>	Enters configuration mode.
<b>Step 2</b>	<code>policy-map type qos qos-policy-map-name</code>	Creates or accesses the policy map named policy-map-name, and then enters policy-map mode. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
<b>Step 3</b>	<code>class [type qos] {class-map-name   class-default}</code>	Creates a reference to class-map-name, and enters policy-map class configuration mode. Use the <b>class-default</b> keyword to select all traffic that is not currently matched by classes in the policy map.
<b>Step 4</b>	<code>set dscp dscp-value</code>	Sets the DSCP value to dscp-value. See the Standards DSCP Values table.

This example shows how to display the policy-map configuration as shown below:

```
switch# show policy-map policy1
```

## Configuring IP Precedence Marking

You can set the value of the IP precedence field in bits 0 to 2 of the IPv4 type of service (ToS) field or the equivalent Traffic Class field for IPv6 of the IP header. The following table shows the precedence values:



**Note** You can set IP Precedence or DSCP but you can not set both values because they modify the same field in the IP packet.

**Table 5: Precedence Values**

Value	List of Precedence Values
<0-7>	IP precedence value
critical	Critical precedence (5)
flash	Flash precedence (3)



Value	List of Precedence Values
flash-override	Flash override precedence (4)
immediate	Immediate precedence (2)
internet	Internetwork control precedence (6)
network	Network control precedence (7)
priority	Priority precedence (1)
routine	Routine precedence (0)

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<code>confi t</code>	Enters configuration mode.
<b>Step 2</b>	<code>policy-map [type qos] qos-policy-map-name</code>	Creates or accesses the policy map named policy-map-name, and then enters policy-map mode. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
<b>Step 3</b>	<code>class [type qos] {class-map-name   class-default}</code>	Creates a reference to class-map-name, and enters policy-map class configuration mode. Use the <b>class-default</b> keyword to select all traffic that is not currently matched by classes in the policy map.
<b>Step 4</b>	<code>set precedence precedence-value</code>	Sets the IP precedence value to precedence-value. You can enter one of the values shown in the Precedence Values table.

```
switch(config)# policy-map type qos my_policy
switch(config-pmap-qos)# class type qos my_class
switch(config-pmap-c-qos)# set precedence 5
switch(config-pmap-c-qos)#
```

## QoS Configurations for Layer 3 Routing

### Required CoS Marking Configuration in a Layer 3 Topology

In Layer 3 topologies, you must configure each qos-group in the network-qos policy with a unique cos value.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>show policy-map system</b>	Displays the already configured policy-maps and CoS values.  In Layer 3 topologies, each qos-group must have a unique CoS value. Use the <b>show policy-map system</b> command to view CoS values that have been used and that are unavailable for qos-groups.
<b>Step 2</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 3</b>	switch(config) # <b>policy-map</b> [ <b>type network-qos</b> ] <i>policy-map name</i>	Creates or accesses the policy map named policy-map-name and enters policy-map mode.  The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
<b>Step 4</b>	switch(config-pmap-nq) # <b>class</b> [ <b>type network-qos</b> ] { <i>class-map name</i>   <b>class-default</b> }	Creates a reference to class-map-name and enters policy-map class configuration mode.  Use the <b>class-default</b> keyword to select all traffic that is not currently matched by classes in the policy map.
<b>Step 5</b>	switch(config-pmap-nq-c) # <b>set cos</b> <i>cos-value</i>	Specifies the CoS value.  The value can range from 0 to 7.  <b>Note</b> You can use this command only in egress policies.  In Layer 3 topologies, each qos-group must have a unique cos configuration.

The following example shows how to set the CoS value to 4 in a Layer 3 topology:

```
switch# show policy-map system
Type network-qos policy-maps
=====

policy-map type network-qos pn-01
  class type network-qos cn-01      match qos-group 1
    mtu 8500
    pause no-drop
    set cos 2
  class type network-qos cn-02      match qos-group 2
    set cos 4
    mtu 9216
  class type network-qos cn-03      match qos-group 3
    mtu 8000
    set cos 6
  class type network-qos cn-04      match qos-group 4
    mtu 8750
    set cos 7
  class type network-qos cn-ip-multicast match qos-group 5
    set cos 5
    mtu 7500
  class type network-qos class-default match qos-group 0
    mtu 1500
    multicast-optimize
    set cos 1
```

```

...
switch# configure terminal
switch(config)# policy-map type network-qos pn-01
switch(config-pmap-nq)# class type network-qos cn-05
switch(config-pmap-c-nq)# set cos 3

```

## Configuring Layer 3 Multicast Queuing

You can map CoS values to an egress queue. You can have a maximum of 4 multicast queues for Layer 3 multicast traffic.

You can use this procedure to distribute traffic into different queues, where each queue is configured with different weighted round robin (WRR) parameters.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>wrr-queue qos-group-map</b> <i>queue-id group1 ...group8</i>	<p>Maps the assigned CoS values to an egress queue.</p> <p>The egress queue range is from 1 to 4, where 4 can be configured as the expedite queue.</p> <p>You can enter up to eight CoS values. Separate each value with a space. The range is from 0 to 7.</p> <p>The defaults are as follows:</p> <ul style="list-style-type: none"> <li>• Receive queue 0 and transmit queue 0: CoS 0 and 1.</li> <li>• Receive queue 1 and transmit queue 1: CoS 2 and 3.</li> <li>• Receive queue 2 and transmit queue 2: CoS 4 and 5.</li> <li>• Receive queue 3 and transmit queue 3: CoS 6 and 7.</li> </ul>

This example shows how to configure a Layer 3 interface:

```

switch# configure terminal
switch(config)# wrr-queue qos-group-map 1 5
switch(config)#

```

## Configuring a Service Policy for a Layer 3 Interface

You can configure a service policy to a Layer 3 interface.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.

	Command or Action	Purpose
<b>Step 2</b>	switch(config)# <b>interface ethernet slot/port</b>	Enters configuration mode for the specified interface.
<b>Step 3</b>	switch(config-if)# <b>no switchport</b>	Selects the Layer 3 interface.
<b>Step 4</b>	switch(config-if)# <b>service-policy [type {qos   queuing output} policy-name</b>	Specifies the policy map to use as the service policy for the Layer 3 interface. There are two policy-map configuration modes: <ul style="list-style-type: none"> <li>• qos—Classification mode; this is the default mode.</li> <li>• queuing—Queuing mode.</li> </ul> <p><b>Note</b> The <b>output</b> keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply <b>output</b> to a queuing policy.</p>

This example shows how to attach a queuing policy map to a Layer 3 interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/5
switch(config-if)# no switchport
switch(config-if)# service-policy type queuing output my_output_q_policy
switch(config-if)#
```

## Changing the Bandwidth Allocated to Unicast and Multicast Traffic

You can change the bandwidth allocated to unicast and multicast traffic by assigning weighted round robin (WRR) weights as a percentage of the interface data rate to the egress queues.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>interface ethernet slot/port</b>	Enters configuration mode for the specified interface.
<b>Step 3</b>	switch(config-if)# <b>wrr unicast-bandwidth percentage-value</b>	Changes the bandwidth allocated to unicast and multicast traffic on traffic congestion. The percent bandwidth value ranges from 0 to 100 percent.

This example shows how to attach a queuing policy map to a Layer 3 interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/5
switch(config-if)# wrr unicast-bandwidth 75
switch(config-if)#
```

## Attaching the System Service Policy

The `service-policy` command specifies the system class policy map as the service policy for the system.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>system qos</b>	Enters system class configuration mode.
<b>Step 3</b>	switch(config-sys-qos)# <b>service-policy type</b> { <b>network-qos</b> [ <b>input</b>   <b>output</b> ]   <b>qos input</b>   <b>queuing output</b> } <i>policy-name</i>	Specifies the policy map to use as the service policy for the system. There are three policy-map configuration modes: <ul style="list-style-type: none"> <li>• <b>network-qos</b>—Network-wide (system qos) mode.</li> <li>• <b>qos</b>—Classification mode (system qos input or interface input only).</li> <li>• <b>queuing</b>—Queuing mode (output at system qos and interface).</li> </ul> <p><b>Note</b> There is no default policy-map configuration mode; you must specify the <b>type</b>. The <b>input</b> keyword specifies that this policy map should be applied to traffic received on an interface. The <b>output</b> keyword specifies that this policy-map should be applied to traffic transmitted from an interface. You can only apply <b>input</b> to a qos policy; you can only apply <b>output</b> to a queuing policy.</p>

This example shows how to set a no-drop Ethernet policy map as the system class:

## Restoring the Default System Service Policies

If you have created and attached new policies to the system QoS configuration, enter the **no** form of the command to reapply the default policies.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>system qos</b>	Enters system class configuration mode.
<b>Step 3</b>	switch(config-sys-qos)# <b>no service-policy type</b> <b>qos input</b> <i>policy-map name</i>	Resets the classification mode policy map. This policy-map configuration is for system QoS input or interface input only:

	Command or Action	Purpose
<b>Step 4</b>	<code>switch(config-sys-qos)# no service-policy type network-qos <i>policy-map name</i></code>	Resets the network-wide policy map.
<b>Step 5</b>	<code>switch(config-sys-qos)# no service-policy type queuing output <i>policy-map name</i></code>	Resets the output queuing mode policy map.

This example shows how to reset the system QoS configuration:

```
switch# configure terminal
switch(config)# system qos
switch(config-sys-qos)# no service-policy type qos input my-in-policy
switch(config-sys-qos)# no service-policy type network-qos my-nq-policy
switch(config-sys-qos)# no service-policy type queuing output my-out-policy
```

This example shows the default service policies:

## Enabling the Jumbo MTU

You can enable the jumbo MTU for the whole switch by setting the MTU to its maximum size (9216 bytes) in the policy map for the default Ethernet system class (class-default).



### Note

The Cisco Nexus 3000 Series supports 1 MTU for all classes for all ports.

This example shows how to configure the default Ethernet system class to support the jumbo MTU:

```
switch(config)# policy-map type network-qos jumbo
switch(config-pmap-nq)# class type network-qos class-default
switch(config-pmap-c-nq)# mtu 9216
switch(config-pmap-c-nq)# exit
switch(config-pmap-nq)# exit
switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos jumbo
```



### Note

The **system jumbomtu** command defines the maximum MTU size for the switch. However, jumbo MTU is only supported for system classes that have MTU configured.

## Verifying the Jumbo MTU

To verify that the jumbo MTU is enabled, enter the **show interface ethernet *slot/port*** command for an Ethernet interface that carries traffic with jumbo MTU.

This example shows how to display summary jumbo MTU information for Ethernet 1/2 (the relevant part of the output is shown in bold font):

```
switch# show interface ethernet 1/2
Ethernet1/2 is up
...
Rx
1547805598 Input Packets 1547805596 Unicast Packets 0 Multicast Packets
0 Broadcast Packets 1301767362 Jumbo Packets 33690 Storm Suppression Packets
7181776513802 Bytes
```

```
Tx
1186564478 Output Packets 7060 Multicast Packets
0 Broadcast Packets 997813205 Jumbo Packets
4813632103603 Bytes
```

.....  
This example shows how to display detailed jumbo MTU information for Ethernet 1/2 (the relevant part of the output is shown in bold font):

```
switch# show interface ethernet 1/2 counters detailed
Rx Packets: 1547805598
Rx Unicast Packets: 1547805596
Rx Jumbo Packets: 1301767362
Rx Bytes: 7181776513802
Rx Storm Suppression: 33690
Rx Packets from 0 to 64 bytes: 169219
Rx Packets from 65 to 127 bytes: 10657133
Rx Packets from 128 to 255 bytes: 21644488
Rx Packets from 256 to 511 bytes: 43290596
Rx Packets from 512 to 1023 bytes: 86583071
Rx Packets from 1024 to 1518 bytes: 83693729
Rx Trunk Packets: 1547805596
Tx Packets: 1186564481
Tx Unicast Packets: 1005445334
Tx Multicast Packets: 7063
Tx Jumbo Packets: 997813205
Tx Bytes: 4813632103819
Tx Packets from 0 to 64 bytes: 137912
Tx Packets from 65 to 127 bytes: 8288443
Tx Packets from 128 to 255 bytes: 16596457
Tx Packets from 256 to 511 bytes: 33177999
Tx Packets from 512 to 1023 bytes: 66363944
Tx Packets from 1024 to 1518 bytes: 64186521
Tx Trunk Packets: 1005451729
```

## Configuring QoS on Interfaces

### Configuring Untagged CoS

Any incoming packet not tagged with an 802.1p CoS value is assigned the default untagged CoS value of zero (which maps to the default Ethernet drop system class). You can override the default untagged CoS value for an Ethernet or EtherChannel interface.

#### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>interface {ethernet [chassis/]slot/port   port-channel channel-number}</b>	Enters configuration mode for the specified interface or port channel.
<b>Step 3</b>	switch(config-if)# <b>no switchport</b>	(Optional) Selects a Layer 3 interface.
<b>Step 4</b>	switch(config-if)# <b>untagged cos cos-value</b>	Configures the untagged CoS value. Values can be from 1 to 7.

This example shows how to set the CoS value to 4 for untagged frames received on an interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# untagged cos 4
```

This example shows how to set the CoS value to 3 for untagged frames received on a Layer 3 interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/5
switch(config-if)# no switchport
switch(config-if)# untagged cos 3
switch(config-if)#
```

## Configuring Interface Service Policy

An input qos policy is a service policy applied to incoming traffic on an Ethernet interface for classification. For type queuing, the output policy is applied to all outgoing traffic that matches the specified class.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters configuration mode.
<b>Step 2</b>	switch(config)# <b>interface</b> { <b>ethernet</b> [ <i>chassis</i> ]/ <i>slot/port</i>   <b>port-channel</b> <i>channel-number</i> }	Enters configuration mode for the specified interface. <b>Note</b> The service policy on a port channel applies to all member interfaces.
<b>Step 3</b>	switch(config-if)# <b>service-policy</b> [ <b>type</b> { <b>qos input</b>   <b>queuing</b> <b>output</b> }] <i>policy-name</i>	Specifies the policy map to use as the service policy for the system. There are two policy-map configuration modes: <ul style="list-style-type: none"> <li>• qos—Classification mode; this is the default mode.</li> <li>• queuing—Queuing mode.</li> </ul> <b>Note</b> The <b>input</b> keyword specifies that this policy map should be applied to traffic received on an interface. The <b>output</b> keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply <b>input</b> to a qos policy; you can only apply <b>output</b> to a queuing policy.
<b>Step 4</b>	switch(config-if)# <b>service-policy</b> <b>input</b> <i>policy-name</i>	Applies the policy map to the interface. <b>Note</b> There is a restriction that system type qos policy cannot be the same as any the type qos policy applied to an interface or EtherChannel.

This example shows how to apply a policy to an Ethernet interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/1
switch(config-if)# service-policy type qos input policy1
```



## Verifying QoS Configuration

To verify Cisco Nexus 3000 Series QoS configurations, perform one of these tasks:

Command	Purpose
switch# <b>show class-map</b>	Displays the class maps defined on the switch.
switch# <b>show policy-map</b> [ <i>name</i> ]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
switch# <b>show policy-map interface</b> [ <i>interface number</i> ]	Displays the policy map settings for an interface or all interfaces.
switch# <b>show policy-map system</b>	Displays the policy map settings attached to the system qos.
switch# <b>show policy-map type</b> { <i>network-qos</i>   <i>qos</i>   <i>queuing</i> } [ <i>name</i> ]	Displays the policy map settings for a specific policy type. Optionally, you can display the named policy only.
switch# <b>show interface untagged-cos</b> [ <i>module number</i> ]	Displays the untagged CoS values for all interfaces.
switch# <b>show wrr-queue cos-map</b> [ <i>var</i> ]	Displays the mapped CoS values to egress queues.
switch# <b>running-config ipqos</b>	Displays information about the running configuration for QoS.
switch# <b>startup-config ipqos</b>	Displays information about the startup configuration for QoS.
switch# <b>show queuing interface ethernet</b> <i>slot-no/port-no</i>	Displays the queuing information on interfaces.

This example shows how to configure a network qos policy:

```
switch(config)# class-map type network-qos cnq1
switch(config-cmap-nq)# match qos-group 1
switch(config-cmap-nq)# exit
switch(config)# class-map type network-qos cnq6
switch(config-cmap-nq)# match qos-group 6
switch(config-cmap-nq)# exit
switch(config)# policy-map type network-qos pnqos
switch(config-pmap-nq)# class type network-qos cnq1
switch(config-pmap-nq-c)# set cos 4
switch(config-pmap-nq-c)# exit
switch(config-pmap-nq)# class type network-qos cnq6
switch(config-pmap-nq-c)# set cos 5
switch(config-pmap-nq-c)# congestion-control random-detect ecn
switch(config-pmap-nq-c)# exit
switch(config-pmap-nq)# class type network-qos class-default
switch(config-pmap-nq-c)# mtu 9216
switch(config-pmap-nq-c)# exit
switch(config-pmap-nq)# exit
```

```
switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos pnqos
switch(config-sys-qos)#
```

This example shows how to configure a queuing policy:

```
switch(config)# class-map type queuing cq1
switch(config-cmap-que)# match qos-group 1
switch(config-cmap-que)# exit
switch(config)# class-map type queuing cq6
switch(config-cmap-que)# match qos-group 6
switch(config-cmap-que)# exit
switch(config)# policy-map type queuing pqu
switch(config-pmap-que)# class type queuing class-default
switch(config-pmap-c-que)# bandwidth percent 70
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# class type queuing cq1
switch(config-pmap-c-que)# bandwidth percent 10
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# class type queuing cq6
switch(config-pmap-c-que)# bandwidth percent 20
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# exit
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing output pqu
switch(config-sys-qos)#
```

This example shows how to configure a QoS policy:

```
switch(config)# class-map type qos cqos1
switch(config-cmap-qos)# match cos 1
switch(config-cmap-qos)# exit
switch(config)# class-map type qos cqos6
switch(config-cmap-qos)# match cos 6
switch(config-cmap-qos)# exit
switch(config)# policy-map type qos pqos
switch(config-pmap-qos)# class type qos cqos1
switch(config-pmap-c-qos)# set qos-group 1
switch(config-pmap-c-qos)# exit
switch(config-pmap-qos)# class type qos cqos6
switch(config-pmap-c-qos)# set qos-group 6
switch(config-pmap-c-qos)# exit
switch(config-pmap-qos)# exit
switch(config)# system qos
switch(config-sys-qos)# service-policy type qos input pqos
switch(config-sys-qos)#
```

This example shows how to verify the untagged-cos configuration on interfaces:

```
switch(config-if)# show interface untagged-cos
=====
```

```
Interface      Untagged-CoS
=====
```

```
Ethernet1/1  4
Ethernet1/2
Ethernet1/3  5
Ethernet1/4
Ethernet1/5
Ethernet1/6
Ethernet1/7
Ethernet1/8
Ethernet1/9
Ethernet1/10
Ethernet1/11
Ethernet1/12
Ethernet1/13
Ethernet1/14
Ethernet1/15
Ethernet1/16
Ethernet1/17
```

This example shows how to display the QoS running configuration:

```
switch(config)# show running-config ipqos
```

```

!Command: show running-config ipqos
!Time: Mon Mar 15 08:24:12 2010

version 5.0(3)U1(1)
class-map type qos match-all cqos1
  match cos 1
class-map type qos match-all cqos6
  match cos 6
class-map type queuing cqul
  match qos-group 1
class-map type queuing cqu6
  match qos-group 6
policy-map type qos pqos
  class cqos1
    set qos-group 1
  class cqos6
    set qos-group 6
policy-map type queuing pqu
  class type queuing cqul
    bandwidth percent 10
  class type queuing cqu6
    bandwidth percent 20
  class type queuing class-default
    bandwidth percent 70
class-map type network-qos cnq1
  match qos-group 1
class-map type network-qos cnq6
  match qos-group 6
policy-map type network-qos pnqos
  class type network-qos cnq1
    set cos 4
  class type network-qos cnq6
    set cos 5
    congestion-control random-detect ecn
  class type network-qos class-default
    mtu 9216
system qos
  service-policy type qos input pqos
  service-policy type network-qos pnqos
  service-policy type queuing output pqu
    
```

```

interface Ethernet1/1
  untagged cos 4

interface Ethernet1/3
  untagged cos 5
    
```

switch(config)#  
**This example shows how to display the QoS groups that are mapped to the egress queue:**

```

switch(config)# wrq-queue qos-group-map 3 1
switch(config)# show wrq-queue qos-group-map
MCAST Queue ID      Qos-Group Map
0                    0
1                    2 3
2                    4 5
3                    1 6 7
    
```

switch(config)#  
**This example shows the class-map configuration:**

```

switch(config)# show class-map

Type qos class-maps
=====

class-map type qos match-all cqos1
  match cos 1

class-map type qos match-all cqos6
  match cos 6
    
```

```
class-map type qos match-any class-default
  match any
```

```
Type queuing class-maps
```

```
=====
```

```
class-map type queuing cq1
  match qos-group 1
```

```
class-map type queuing cq6
  match qos-group 6
```

```
class-map type queuing class-default
  match qos-group 0
```

```
Type network-qos class-maps
```

```
=====
```

```
class-map type network-qos cnq1
  match qos-group 1
```

```
class-map type network-qos cnq6
  match qos-group 6
```

```
class-map type network-qos class-default
  match qos-group 0
```

```
switch(config)#
```

This example shows the policy-map configuration:

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

```
Type qos policy-maps
```

```
=====
```

```
policy-map type qos pqos
  class type qos cqos1
    set qos-group 1
  class type qos cqos6
    set qos-group 6
  class type qos class-default
    set qos-group 0
policy-map type qos default-in-policy
  class type qos class-default
    set qos-group 0
```

```
Type queuing policy-maps
```

```
=====
```

```
policy-map type queuing pqu
  class type queuing cq1
    bandwidth percent 10
  class type queuing cq6
    bandwidth percent 20
  class type queuing class-default
    bandwidth percent 70
policy-map type queuing default-out-policy
  class type queuing class-default
    bandwidth percent 100
```

```
Type network-qos policy-maps
```

```
=====
```

```
policy-map type network-qos pnqos
  class type network-qos cnq1
    mtu 1500
    set cos 4
  class type network-qos cnq6
```

```

        mtu 1500
        set cos 5
        congestion-control random-detect ecn
    class type network-qos class-default
        mtu 9216
policy-map type network-qos default-nq-policy
    class type network-qos class-default
        mtu 1500
switch(config)#

```

This example shows all active policy maps in the system:

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

```

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

policy-map type network-qos pnqos
  class type network-qos cnq1      match qos-group 1

      mtu 1500
      set cos 4
  class type network-qos cnq6      match qos-group 6

      mtu 1500
      set cos 5
      congestion-control random-detect ecn
  class type network-qos class-default      match qos-group 0

      mtu 9216

Service-policy (qos) input:  pqos
policy statistics status:   disabled

Class-map (qos):  cqos1 (match-all)
  Match: cos 1
  set qos-group 1

Class-map (qos):  cqos6 (match-all)
  Match: cos 6
  set qos-group 6

Class-map (qos):  class-default (match-any)
  Match: any
  set qos-group 0

Service-policy (queuing) output:  pqu
policy statistics status:   disabled

Class-map (queuing):  cq1 (match-any)
  Match: qos-group 1
  bandwidth percent 10

Class-map (queuing):  cq6 (match-any)
  Match: qos-group 6
  bandwidth percent 20

Class-map (queuing):  class-default (match-any)
  Match: qos-group 0
  bandwidth percent 70

```

```
switch(config)#
```

This example shows the service policy maps configured on the interfaces:

```
switch(config)# show policy-map interface ethernet 1/1
```

```
Global statistics status :  disabled
```

```
Ethernet1/1
```

```

Service-policy (qos) input:  pqos
policy statistics status:   disabled

```

```

Class-map (qos):  cqos1 (match-all)
  Match: cos 1
  set qos-group 1

Class-map (qos):  cqos6 (match-all)
  Match: cos 6
  set qos-group 6

Class-map (qos):  class-default (match-any)
  Match: any
  set qos-group 0

Service-policy (queuing) output:  pqu
  policy statistics status:  disabled

Class-map (queuing):  cqul (match-any)
  Match: qos-group 1
  bandwidth percent 10

Class-map (queuing):  cqu6 (match-any)
  Match: qos-group 6
  bandwidth percent 20

Class-map (queuing):  class-default (match-any)
  Match: qos-group 0
  bandwidth percent 70

switch(config)#
This example shows the queuing information for a specific interface:

```

```

switch(config)# show queuing interface ethernet 1/1
Ethernet1/1 queuing information:
  TX Queuing
    qos-group  sched-type  oper-bandwidth
      0         WRR        70
      1         WRR        10
      6         WRR        20

  RX Queuing
    qos-group 0
    HW MTU: 1500 (1500 configured)
    drop-type: drop, xon: 0, xoff: 0
    Statistics:
      Ucast pkts sent over the port      : 0
      Ucast bytes sent over the port     : 0
      Mcast pkts sent over the port      : 0
      Mcast bytes sent over the port     : 0
      Ucast pkts dropped                 : 0
      Ucast bytes dropped                 : 0
      Mcast pkts dropped                 : 0
      Mcast bytes dropped                 : 0
    qos-group 1
    HW MTU: 1500 (1500 configured)
    drop-type: drop, xon: 0, xoff: 0
    Statistics:
      Ucast pkts sent over the port      : 0
      Ucast bytes sent over the port     : 0
      Mcast pkts sent over the port      : 0
      Mcast bytes sent over the port     : 0
      Ucast pkts dropped                 : 0
      Ucast bytes dropped                 : 0
      Mcast pkts dropped                 : 0
      Mcast bytes dropped                 : 0
    qos-group 6
    HW MTU: 1500 (1500 configured)
    drop-type: drop, xon: 0, xoff: 0
    Statistics:
      Ucast pkts sent over the port      : 0
      Ucast bytes sent over the port     : 0
      Mcast pkts sent over the port      : 0
      Mcast bytes sent over the port     : 0
      Ucast pkts dropped                 : 0

```

```

Ucast bytes dropped           : 0
Mcast pkts dropped           : 0
Mcast bytes dropped           : 0
switch(config)#

```

## Monitoring the QoS Packet Buffer

The Cisco Nexus 3000 series has a 9MB buffer memory which is divided into dedicated per port and dynamic shared memory. Each front-panel port has 8 unicast and 4 multicast queues in egress.

In the scenario of burst or congestion, each egress port consumes buffers from the dynamic shared memory.

Beginning with Cisco NX-OS Release 5.0(3)U2(1), you can display real time status of the shared buffer on a per port basis. All counters are displayed in terms of the number of cells. Each cell is 208 bytes in size. You can also display the global level buffer consumption in terms of consumption and available number of cells.

This example shows how to:

```
switch(config)# show hardware internal buffer info pkt-stats
```

```

-----|
Total Instant Usage           7588
Remaining Instant Usage       38492
Max Cell Usage                7945
Switch Cell Count             46080
-----|
Instant Buffer utilization per queue per port
Each line displays the number of cells utilized for a given port for each QoS queue
One cell represents approximately 208 bytes
-----+-----+-----+-----+-----+-----+-----+-----+
Port      Q1      Q2      Q3      Q4      Q5      Q6      Q7      Q8
-----+-----+-----+-----+-----+-----+-----+-----+
[ 6]
UC->      0      0      0      0      0      0      0      0
MC->      4      0      0      0
[ 9]
UC->      0      0      0      0      0      0      0      0
MC->    3807      0      0      0
[13]
UC->      0      0      0      0      0      0      0      0
MC->      4      0      0      0
[19]
UC->      0      0      0      0      0      0      0      0
MC->    3802      0      0      0

```

Usage information:

- Total Instant Usage---Current buffer usage in terms of the number of cells on a global basis.
- Remaining Instant Usage---The effective free number of cells available on a global basis.
- Max Cell Usage---The maximum buffer usage that is seen until the last clear.
- Switch Cell Count---Total global buffer space available in the platform in terms of the number of cells on a global basis.

UC and MC represent the 8 unicast (Q1-Q8) and 4 multicast (Q1-Q4) instant cell usage. The example above shows the multicast queue Q1 is consuming 3807 cells instantaneously on port 9.

This example shows how to clear system buffer max cell usage counter:

```
switch# clear counters buffers  
Max Cell Usage has been reset successfully
```

This example shows how to set a buffer utilization threshold on a per port basis. If the buffer occupancy exceeds this number, you can generate a syslog or check the status in the **show hardware internal buffer info pkt-stats port-log** command:

```
switch# hardware profile buffer info port-threshold front-port 1 threshold 10  
Port threshold changed successfully
```

This example displays the last time when the buffer utilization on this port exceeded the configured threshold value:

```
switch(config)# sh hardware internal buffer info pkt-stats port-log  
02-27-2012 04:10:36.63345 Port 9 buffer threshold 3685 exceeded 810[3%]  
02-27-2012 04:10:36.63764 Port 17 buffer threshold 3684 exceeded 2430[9%]  
02-27-2012 04:10:36.65436 Port 63 buffer threshold 3681 exceeded 270[1%]
```





## INDEX

### A

- attaching [29](#)
  - system service policy [29](#)

### B

- bandwidth [28](#)
  - multicast traffic [28](#)
  - unicast traffic [28](#)

### C

- changed information [1](#)
  - description [1](#)
- class maps [11](#)
  - configuring [11](#)
- classification policies [10](#)
  - ingress [10](#)
- CoS marking [22, 25](#)
  - configuring [22](#)
  - layer 3 [25](#)
- CPU traffic [10](#)
  - QoS [10](#)

### D

- default system service policies [29](#)
  - restoring [29](#)
- DSCP classification [14](#)
  - configuring [14](#)

### I

- information about [6, 7](#)
  - policy types [7](#)
  - system classes [6](#)

- ingress [10](#)
  - classification policies [10](#)
- IP precedence marking [24](#)
  - qos [24](#)

### L

- layer 3 [25](#)
  - CoS marking [25](#)

### M

- modular QoS CLI [5](#)
- MQC [5](#)
- multicast traffic [28](#)
  - changing bandwidth [28](#)

### N

- new information [1](#)
  - description [1](#)

### O

- overview [5](#)
  - quality of service [5](#)

### P

- packet buffer, monitoring [39](#)
  - qos [39](#)
- policy maps [17](#)
  - creating [17](#)
- policy types [7](#)
  - information about [7](#)

precedence classification [16](#)  
  configuring [16](#)

## Q

qos [24](#)  
  IP precedence marking [24](#)  
Qos [10](#)  
  CPU traffic [10](#)  
QoS [14, 16](#)  
  DSCP classification [14](#)  
    configuring [14](#)  
  precedence classification [16](#)  
    configuring [16](#)  
quality of service [5](#)  
  overview [5](#)  
Quality of Service, See [QoS](#)

## R

restoring [29](#)  
  default system service policies [29](#)

## S

system classes [6](#)  
  information about [6](#)  
system service policy [29](#)  
  attaching [29](#)

## T

type QoS polices, configuring [19](#)

## U

unicast traffic [28](#)  
  changing bandwidth [28](#)