



## **Cisco Nexus 3000 Series NX-OS QoS Configuration Guide, Release 6.x**

**First Published:** 2013-05-21

**Last Modified:** 2020-08-20

### **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-29549-02

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)

© 2011–2020 Cisco Systems, Inc. All rights reserved.



## CONTENTS

<b>CHAPTER 1</b>	<b>New and Changed Information 1</b>
	New and Changed Information 1
<b>CHAPTER 2</b>	<b>Overview 3</b>
	Quality of Service Overview 3
<b>CHAPTER 3</b>	<b>Configuring QoS 5</b>
	Information About Quality of Service 5
	Modular QoS CLI 5
	System Classes 6
	Default System Classes 6
	Information About Policy Types 7
	MTU 11
	Trust Boundaries 12
	Ingress Classification Policies 12
	Priority Groups for No-Drop Classes 13
	Egress Queuing Policies 13
	QoS for Traffic Directed to the CPU 14
	QoS Configuration Guidelines and Limitations 14
	Configuring System Classes 16
	Configuring Class Maps 16
	Configuring ACL Classification 17
	Configuring CoS Classification 18
	Configuring DSCP Classification 19
	Configuring IP RTP Classification 20
	Configuring Precedence Classification 21

Creating Policy Maps	22
Configuring Type QoS Policies	24
Configuring Type Network QoS Policies	25
Configuring Type Queuing Policies	26
Configuring an ECN Threshold	29
Configuring Pause Buffer Thresholds and Priority Groups	32
Information About Marking	34
Configuring CoS Marking	34
Configuring DSCP Marking	35
Configuring IP Precedence Marking	36
QoS Configurations for Layer 3 Routing	38
Configuring Layer 3 Multicast Queuing	38
Configuring a Service Policy for a Layer 3 Interface	38
Changing the Bandwidth Allocated to Unicast and Multicast Traffic	39
Attaching the System Service Policy	40
Restoring the Default System Service Policies	41
Enabling the Jumbo MTU	41
Verifying the Jumbo MTU	42
Configuring QoS on Interfaces	47
Configuring Untagged CoS	47
Configuring an Interface Service Policy	48
Verifying the QoS Configuration	49
Monitoring the QoS Packet Buffer	58

---

**CHAPTER 4**

<b>Configuring Priority Flow Control</b>	<b>61</b>
About Priority Flow Control	61
Guidelines and Limitations for Priority Flow Control	62
Default Settings for Priority Flow Control	63
Enabling Priority Flow Control on a Traffic Class	64
Configuring Priority Flow Control	66
Reserving mmu-buffer for PFC	66
Configuring a Priority Flow Control Watchdog Interval	67
Verifying the Priority Flow Control Configuration	69
Monitoring PFC Frame Counter Statistics	69

Configuration Examples for Priority Flow Control 70

---

**CHAPTER 5****Configuring Traffic Shaping 71**

About Traffic Shaping 71

Guidelines and Limitations for Traffic Shaping 72

Configuring Traffic Shaping 72

Verifying Traffic Shaping 73

Configuration Example for Traffic Shaping 73





## CHAPTER 1

# New and Changed Information

This chapter contains the following sections:

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

## New and Changed Information

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 guide or of the new features in this release.

Feature	Description	Added or Changed in Release	Where Documented
Enhancement to support zero queue-size to drop all packets on the queue.	The queue-limit CLI under the queuing policy is enhanced to support zero egress queue size to drop all packets on the queue.	6.0(2)U5(1)	<a href="#">QoS Configuration Guidelines and Limitations, on page 14</a> <a href="#">Configuring Type Queuing Policies, on page 26</a>
Burst traffic handling improvement	With the use of buffer pools and buffer modes, buffer handling is enhanced to have better burst absorption capability.	6.0(2)U5(1)	<a href="#">QoS Configuration Guidelines and Limitations, on page 14</a> <a href="#">Monitoring the QoS Packet Buffer, on page 58</a>
Queuing Information for Interfaces	The show queuing command was introduced.	6.0(2)U4(1)	<a href="#">Verifying the QoS Configuration, on page 49</a>
Priority Flow Control	Priority Flow Control (PFC) is a mechanism that prevents frame loss due to congestion.	6.0(2)U2(1)	<a href="#">Configuring Priority Flow Control, on page 61</a>

Feature	Description	Added or Changed in Release	Where Documented
QoS Enhancements	<p>The following enhancements were made to QoS:</p> <ul style="list-style-type: none"><li>• Three strict priority classes</li><li>• New buffer statistics commands</li><li>• ECN enhancements</li><li>• Minimum guaranteed bandwidth in the <b>shape</b> command</li><li>• Queue limit configuration</li><li>• QoS group mapping to priority group</li></ul>	6.0(2)U2(1)	<a href="#">Configuring QoS, on page 5</a>





## CHAPTER 2

# Overview

---

This chapter contains the following sections:

- [Quality of Service Overview, on 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 at egress. At ingress, they are used to configure buffer threshold and priority group mapping.
- Network QoS policies include configuring maximum transmission unit (MTU), pause no-drop, and queue-limit. It is also used to configure explicit congestion notification (ECN) and weighted random early detection (WRED).
- Priority flow control.





## CHAPTER 3

# Configuring QoS

This chapter contains the following sections:

- [Information About Quality of Service, on page 5](#)
- [QoS Configuration Guidelines and Limitations, on page 14](#)
- [Configuring System Classes, on page 16](#)
- [Configuring QoS on Interfaces, on page 47](#)
- [Verifying the QoS Configuration, on page 49](#)
- [Monitoring the QoS Packet Buffer, on page 58](#)

## Information About 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).

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 class of service (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 at egress and for configuring buffer threshold and priority group mapping at ingress.



#### 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.

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

## Default System Classes

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

There are three policy types:

- Network-qos
- Queuing
- QoS

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 is 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** The Cisco Nexus device supports one MTU for traffic for all classes for all ports. However, you can have different MTUs for different classes. The MTUs are used for PFC buffer calculation.

---

- Set CoS value—This configuration is used to mark 802.1p values for all traffic mapped to this system class.
- Congestion Control WRED—Weighted random early detection (WRED) anticipates and avoids congestion before congestion occurs. WRED drops packets, based on the average queue length that exceeds 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 (ECN) feature, routers and end hosts use this marking as a signal that the network is congested to slow down sending packets. To enable an ECN, use the **congestion-control random-detect ecn** command in the network-qos policy map mode.

ECN is supported on all types of Cisco Nexus 3000 series switches.




---

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

---

- No drop—No drop specifies lossless service for the system class.
- Type queuing—The Cisco Nexus device supports type queuing in the ingress and egress directions. Egress type queuing policies are used to define the scheduling characteristics of the queues. Ingress type queuing policies are used to define the pause buffer thresholds, priority group, and queue limit.




---

**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 is 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.

---

- Egress queuing policy—The egress queuing policy is used to configure egress queues on the device.
  - 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. For Cisco Nexus 3132 and 3172 switches, there are three strict priority levels.
  - Shape and minimum guarantee—Specifies the burst size and minimum guaranteed bandwidth for this queue.
  - Queue limit—Specifies either the static or dynamic queue limit for Cisco Nexus 3100 Series switches. The static queue limit defines the fixed size to which the queue can grow.
- Ingress queuing policy—The ingress queuing policy is used to define the pause buffer thresholds, priority group, and queue limit.
  - Pause buffer threshold—Sets the pause and resume buffer threshold settings for ingress traffic.
  - Priority group—Classifies the traffic and monitors statistics on no-drop classes.
  - Queue limit—Sets the shared buffer usage per priority group.

You can configure the threshold for using shared buffers both at ingress and egress based on the alpha value, which is derived from the index. The index ranges from 0 to 9 for Cisco Nexus 3000 series switches and from 0 to 10 for Cisco Nexus 3100 platform switches. At ingress, the alpha value is used to calculate the per port, per priority group share of the buffers available from the current free pool. At egress, the alpha value is used to calculate the per port, per queue share of the buffers available from the current free pool.

For the Cisco Nexus 3000 series switches, the alpha values are as follows:

Index	Alpha Value
0	1/64
1	1/32
2	1/16
3	1/8
4	1/4
5	1/2
6	1
7	2
8	4
9	8
<b>Note</b> Index 9 is not applicable for Cisco Nexus 34180YC.  Cisco Nexus 34180YC has an index range from 0 to 8.	

For the Cisco Nexus 3100 platform switches, the alpha values are as follows:

Index	Alpha Value
0	1/128
1	1/64
2	1/32
3	1/16
4	1/8
5	1/4
6	1/2
7	1
8	2

Index	Alpha Value
9  <b>Note</b> Index 9 is not applicable for Cisco Nexus 34180YC.  Cisco Nexus 34180YC has an index range from 0 to 8.	4
10  <b>Note</b> Index 10 is not applicable for Cisco Nexus 34180YC.  Cisco Nexus 34180YC has an index range from 0 to 8.	8

- To determine the default shared alpha value, use one of the following commands:

- For Cisco NX-OS 6.x, use the **test hardware internal bcm-usd bcm-diag-shell** command to determine the default shared alpha value.

For example:

```
switch# test hardware internal bcm-usd bcm-diag-shell
Available Unit Numbers: 0
bcm-shell.0> d chg MMU_THDU_XPIPE_CONFIG_QUEUE
MMU_THDU_XPIPE_CONFIG_QUEUE.mmu0[0]:
<Q_SHARED_LIMIT_CELL=7,Q_SHARED_ALPHA_CELL=7,Q_MIN_LIMIT_CELL=0xb,
Q_LIMIT_DYNAMIC_CELL=1,Q_COLOR_LIMIT_DYNAMIC_CELL=1,DATA=0x00000000c00160007>
```

- For Cisco NX-OS 7.x, use the **bcm-shell module 1** command to determine the default shared alpha value.

For example:

```
switch# bcm-shell module 1
Available Unit Numbers: 0
bcm-shell.0> d chg MMU_THDU_XPIPE_CONFIG_QUEUE
MMU_THDU_XPIPE_CONFIG_QUEUE.mmu0[0]:
<Q_SHARED_LIMIT_CELL=7,Q_SHARED_ALPHA_CELL=7,Q_MIN_LIMIT_CELL=0xb,
Q_LIMIT_DYNAMIC_CELL=1,Q_COLOR_LIMIT_DYNAMIC_CELL=1,DATA=0x00000000c00160007>
```

- In dynamic mode, the amount of shared buffer a port can consume based on the alpha parameter is the queue's threshold: alpha value \* (number of unused cells in the buffer). In general, as the number of unused cells decreases (ie. the buffer becomes fuller), the queue's threshold reduces.
- 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 that corresponds to the system class this traffic flow is mapped to.

The Cisco Nexus 3000 Series switches support:	<ul style="list-style-type: none"> <li>• Eight QoS groups</li> <li>• Eight queues for unicast</li> <li>• Four queues for multicast</li> </ul> <p><b>Note</b> Cisco Nexus 34180YC has eight queues for both unicast and multicast.</p>
The Cisco Nexus 3100 platform switches support:	<ul style="list-style-type: none"> <li>• Eight QoS groups</li> <li>• Eight queues for unicast</li> <li>• Eight queues for multicast</li> </ul> <p>For Cisco Nexus 3100 platform switches, each QoS group is mapped to one multicast queue. The mapping is QoS group 0 mapped to multicast queue 1, QoS group 1 mapped to multicast queue 2, and so forth.</p>

## MTU

The Cisco Nexus device supports one MTU for all classes for all ports.

When configuring MTU, follow these guidelines:

- For the Cisco Nexus device, the MTU is controlled by the value configured on the class default.

- Enter the **system jumbomtu** command to define the upper bound of any MTU in the system. The system jumbo MTU has a default value of 9216 bytes. The minimum MTU is 1500 bytes and the maximum MTU is 9216 bytes.
- The system class MTU sets the MTU for all packets in the class. The system class MTU cannot be configured larger than the global jumbo MTU.
- The default system class has a default MTU of 1500 bytes. You can configure this value.
- You can specify the MTU value for either a single Layer 3 interface or a range of Layer 3 interfaces. When you change the Layer 3 interface MTU value to the jumbo MTU value (1500 bytes or greater), you must also change the network QoS MTU value to 1500 bytes or greater.
- You can set the MTU per class of the network-qos policy. The MTU that is set is used to decide the buffer allocations for PFC. On a need basis, you can configure some classes to have an MTU of 9216 and some to have an MTU of 1500, depending on the type of traffic expected on that class. This will help the system configure the PFC buffers when a class is configured as a no-drop-class.
- On Cisco Nexus 3500 Switches, MTU for all classes must be same as the one configured for the default-class.

## 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.




---

**Note** For Cisco Nexus 34180YC, all ports are trusted interfaces.

---

- 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.

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 packet property (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.

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

## Priority Groups for No-Drop Classes

In Cisco Nexus 3000 series switches and Cisco Nexus 3100 platform switches, packets are handled as cells. Each cell holds 208 bytes of data. One packet can be split into many cells, but each cell can contain a maximum of one packet. Priority groups are groups of cells on which the PFC thresholds are applied. They apply only to no-drop classes and are used for classifying traffic and monitoring statistics.

**Note**

For Cisco Nexus 34180YC, the maximum amount of data in a cell is 80 bytes.

You can associate a no-drop class with a priority group number in the input queuing policy map to guarantee MTU buffers for the specified traffic class. The pause thresholds for the no-drop class are applied on the priority group.

By default, the priority group number is assigned by the system. You can override it by using the **priority-group** command.

**Note**

You cannot have multiple no-drop classes mapped to the same priority group.

## 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.

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

- In addition to these queues, control traffic that is destined for the CPU uses strict priority queues. These queues are not accessible for user configuration.

**Note**

For Cisco Nexus 34180YC, queue 7 is used for CPU traffic and is user configurable. However, as a best practice do not change queue 7 from strict priority to any other type of scheduling. Congesting queue 7 with data traffic leads to control packet loss.

- Standard Ethernet traffic in the default drop system class is assigned a queue. This queue uses WRR scheduling with 100 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 one strict priority queue for Cisco Nexus 3000 series switches and Cisco Nexus 3500 platform switches. This queue is serviced before all other queues except the control traffic queue (which carries control rather than data traffic).




---

**Note** Cisco Nexus 34180YC supports a maximum of seven priority queues.

---

You can configure up to three strict priority queues with multiple priority levels on Cisco Nexus 3100 platform switches.




---

**Note** Cisco Nexus 34180YC supports a maximum of seven priority queues.

---

## 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:

- Starting with Release 7.0(3)I2(1), the **show queuing interface ethernet slot/chassis\_number** command displays two more options: module and summary in the output. These additional options do not have any functionality impact. See the following example of the output:

```
# show queuing interface eth1/1 ?
<CR>
,      Multi range separator
-      Range separator
.      Sub interface separator
>      Redirect it to a file
>>    Redirect it to a file in append mode
module Slot/module
summary Summary
|      Pipe command output to filter
```

- Prior to Release 7.0(3)I2(1), the switch prompt for **policy-map type network-qos** was switch(config-pmap-nq)#. Starting with Release 7.0(3)I2(1), the new switch prompt for **policy-map type network-qos** is switch(config-pmap-nqos)#. The change in the switch prompt is required to be taken care of in the scripts.
- Prior to Release 7.0(3)I2(1), if a non-existing class was configured under a policy-map, an error message would be displayed. Starting with Release 7.0(3)I2(1), if a non-existing class is configured under a policy-map, a new class-map is created and the prompt changes from config-pmap-nqos to config-cmap-nqos as displayed in the following example:

```
switch(config)# show class-map type network-qos

Type network-qos class-maps
=====
class-map type network-qos pfcCos2
  match qos-group 2
class-map type network-qos pfcCos3
```

```

        match qos-group 3
    class-map type network-qos pfcCos5
        match qos-group 5
    class-map type network-qos class-default
        match qos-group 0

switch(config)#
switch(config)# policy-map type network-qos pfcCos
switch(config-pmap-nqos)# class type network-qos pfcCos

switch(config-cmap-nqos)# show class-map type network-qos

Type network-qos class-maps
=====
class-map type network-qos pfcCos

class-map type network-qos pfcCos2
    match qos-group 2
class-map type network-qos pfcCos3
    match qos-group 3
class-map type network-qos pfcCos5
    match qos-group 5
class-map type network-qos class-default
    match qos-group 0

switch(config-cmap-nqos)#

```

- Starting with Release 7.0(3)I2(1), the command **show queuing interface** displays the queues even without applying the network-qos policy.
- The output of the **show queuing interface** command displays an option for the internal HiGig2 interface. These interfaces are not relevant and the additional option does not have any functional impact.
- Starting with Release 7.0(3)I2(1), the output format and the fields of the **show policy-map interface <> type queuing** CLI command have been updated. For Class-map (queuing), the following fields are displayed: policy, bandwidth percent, queue dropped pkts, and queue depth in bytes.
- Prior to Release 7.0(3)I2(1), removing the default bandwidth configuration from the default queuing class used to set the bandwidth to 50%. Starting with Release 7.0(3)I2(1), removing the default bandwidth configuration sets the bandwidth to the default value of 100%. You can set the bandwidth to 50% by configuring **bandwidth percent 50**. You can use the **no bandwidth <bw-input>** CLI command to remove the bandwidth configuration.
- Prior to Release 7.0(3)I2(1), only priority level 1 was supported in the pmap configuration. Starting with Release 7.0(3)I2(1), you can configure priority levels 2 and 3 in the pmap configuration. Release 7.0(3)I2(1) supports only the functionality for priority level 1 for Cisco Nexus 3000 Series platforms even though both priority level 2 and 3 are allowed in the pmap configuration in the Cisco Nexus 3000 Series platforms. Cisco Nexus 3100 Series platforms support priority level 1, 2, 3 with functionality perspective.
- Starting with Release 7.0(3)I2(1), the output format and the fields of the **show queuing interface eth <>** CLI command have been updated. The output displays all qos-groups, control qos group, SPAN qos-group, and pfc statistics. The xon drops, xoff drops, and the HW MTU fields are not displayed in the new format.
- ECN is supported on the Cisco Nexus 3000 series switches and the Cisco Nexus 3132 switch. It is not supported on Cisco Nexus 3172 switches.

- WRED and ECN configuration is not supported on a class mapped to qos-group 1 for Cisco Nexus 3000 Series switches. However, WRED and ECN configuration is supported on a class mapped to qos-group 1 for the Cisco Nexus 3132 switch.
- Starting with Release 6.0(2)U5(1), the queue-limit CLI under the queuing policy is enhanced to support zero egress queue size to drop all packets on the queue. In scenarios where a particular type of traffic needs to be dropped without using the drop ACLs, the traffic can be mapped to a dedicated egress queue and then the queue-size 0 bytes can be applied using this enhancement. As a result, all UC and MC traffic mapped to the queue gets dropped completely.
- Starting with Release 6.0(2)U5(1), the switch allows different modes of min-buffer allocation that can reduce the overall min-buffers allocated per port, in turn increasing the shared pool size. If the shared pool size is higher, the burst absorption capability of the switch is better.

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> [type { <b>network-qos</b>   <b>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>• <b>network-qos</b>—Network-wide (global) mode. CLI prompt: switch(config-cmap-nq)#</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>qos</b>—Classification mode; this is the default mode. CLI prompt: <code>switch(config-cmap-qos)#</code></li> <li>• <b>queuing</b>—Queuing mode. CLI prompt: <code>switch(config-cmap-que)#</code></li> </ul>
<b>Step 3</b>	(Optional) <code>switch(config)# class-map [type qos] [match-all   match-any] class-map name</code>	<p>Specifies that packets must match any or all criteria that is defined for a class map.</p> <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> <p>Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.</p>
<b>Step 4</b>	(Optional) <code>switch(config)# no class-map [type {network-qos   qos   queuing}] class-name</code>	<p>Deletes the specified class map.</p> <p><b>Note</b> You cannot delete the system-defined class map: <code>class-default</code>.</p> <p>Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.</p>

## 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>	<code>switch# configure terminal</code>	Enters global configuration mode.
<b>Step 2</b>	<code>switch(config)# class-map type qos class-name</code>	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.

	Command or Action	Purpose
<b>Step 3</b>	switch(config-cmap-qos)# <b>match 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.
<b>Step 4</b>	(Optional) switch(config-cmap-qos)# <b>no match access-group name</b> <i>acl-name</i>	Removes the match from the traffic class.

### Example

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 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 cos</b> <i>cos-value</i>	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.



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

### Example

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 1: 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

Value	List of DSCP Values
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
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 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 dscp dscp-list</b>	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>	(Optional) switch(config-cmap-qos)# <b>no match dscp dscp-list</b>	Removes the match from the traffic class. For a list of DSCP values, see the Standard DSCP Values table.

### Example

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 global configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map type qos 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 ip rtp port-number</b>	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>	(Optional) switch(config-cmap-qos)# <b>no match ip rtp port-number</b>	Removes the match from the traffic class.

### Example

The following 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 2: 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)

Value	List of Precedence Values
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>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>class-map type qos match-any</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 precedence</b> <i>precedence-values</i>	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>	(Optional) switch((config-cmap-qos)# <b>no match precedence</b> <i>precedence-values</i>	Removes the match from the traffic class. For a list of precedence values, see the Precedence Values table.

### Example

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)# class-map type qos cl
switch(config-cmap-qos)# match precedence 1-2, 5
```

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 device 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
- Input queuing: default-in-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 device 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 global 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>• network-qos—Network-wide (global) mode. CLI prompt: switch(config-pmap-nq)#</li> <li>• qos—Classification mode; this is the default mode. CLI prompt: switch(config-pmap-qos)#</li> <li>• queuing—Queuing mode. CLI prompt: switch(config-pmap-que)#</li> </ul>
<b>Step 3</b>	(Optional) switch(config)# <b>no policy-map</b> [type { <b>network-qos</b>   <b>qos</b>   <b>queuing</b> }] <i>policy-name</i>	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

	Command or Action	Purpose
		<p>system class. The three class-map configuration modes are as follows:</p> <ul style="list-style-type: none"> <li>• network-qos—Network-wide (global) mode. CLI prompt: <code>switch(config-pmap-c-nq)#</code></li> <li>• qos—Classification mode; this is the default mode. CLI prompt: <code>switch(config-pmap-c-qos)#</code></li> <li>• queuing—Queuing mode. CLI prompt: <code>switch(config-pmap-c-que)#</code></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>	(Optional) <code>switch(config-pmap)# no class [type {network-qos   qos   queuing}] class-name</code>	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 eight QoS groups for ingress traffic.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<code>switch# configure terminal</code>	Enters global configuration mode.
<b>Step 2</b>	<code>switch(config)# policy-map type qos policy-name</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-qos)# [class   class-default] type qos class-name</code>	Associates a class map with the policy map, and enters configuration mode for the specified system class. <p><b>Note</b> The associated class map must be the same type as the policy map type.</p>
<b>Step 4</b>	<code>switch(config-pmap-c-qos)# set qos-group qos-group-value</code>	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

	Command or Action	Purpose
		of the <i>qos-group-value</i> . There is no default value.  <b>Note</b> The Cisco 3000 Series Switches supports 8 QoS groups(0-7).  The Cisco 3500 Series Switches supports 4 QoS groups(1-4).
<b>Step 5</b>	(Optional) switch(config-pmap-c-qos)# <b>no set qos-group</b> <i>qos-group-value</i>	Removes the <b>qos-group</b> values from this class.

### Example

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 global configuration mode.
<b>Step 2</b>	switch(config)# <b>policy-map type network-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-nq)# <b>class type network-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-nq)# <b>mtu</b> <i>mtu-value</i>	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.

	Command or Action	Purpose
<b>Step 5</b>	(Optional) <code>switch(config-pmap-c-nq)# no mtu</code>	Resets the MTU value in this class.
<b>Step 6</b>	<code>switch(config-pmap-c-nq)# pause no-drop</code>	Configures a no-drop class.  <b>Note</b> This option works only on class-map matching qos-group 1 in Cisco Nexus 3500 series switches.
<b>Step 7</b>	<code>switch(config-pmap-c-nq)# set cos cos-value</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 8</b>	(Optional) <code>switch(config-pmap-c-nq)# no set cos cos-value</code>	Disables the marking operation in this class.

### Example

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

```
switch# configure terminal
switch(config)# policy-map type network-qos policy-que2
switch(config-pmap-nq)# class type network-qos class-que2
switch(config-pmap-c-nq)# mtu 5000
switch(config-pmap-c-nq)# pause no-drop
switch(config-pmap-c-nq)# congestion-control random-detect
switch(config-pmap-c-nq)# set cos 4
```

## Configuring Type Queuing Policies

Type queuing policies for egress 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>	<code>switch# configure terminal</code>	Enters global configuration mode.
<b>Step 2</b>	<code>switch(config)# policy-map type queuing policy-name</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-que)# class type queuing class-name</code>	Associates a class map with the policy map, and enters configuration mode for the specified system class.
<b>Step 4</b>	<code>switch(config-pmap-c-que)# priority</code>	Specifies that traffic in this class is mapped to a strict priority queue.



	Command or Action	Purpose
		<p><b>Note</b> In Cisco Nexus 3000 series switches and Cisco Nexus 3500 platform switches, only one class in each policy map can have strict priority set on it. Cisco Nexus 3100 platform switches allow multiple priority levels. A maximum of three classes can have priority levels configured. However, only one class can be configured with a particular level. (The Cisco Nexus 34180YC switch supports a maximum of seven priority queues.)</p>
<b>Step 5</b>	(Optional) switch(config-pmap-c-que)# <b>no priority</b>	Removes the strict priority queuing from the traffic in this class.
<b>Step 6</b>	switch(config-pmap-c-que)# <b>shape {kbps   mbps   gbps} burst size min minimum bandwidth</b>	<p>Specifies the burst size and minimum guaranteed bandwidth for this queue.</p> <p><b>Note</b> For the Cisco Nexus 34180YC switch, the command syntax is switch(config-pmap-c-que)# <b>shape {kbps   mbps   gbps} min minimum bandwidth max maximum bandwidth</b>.</p>
<b>Step 7</b>	switch(config-pmap-c-que)# <b>bandwidth percent percentage</b>	<p>Assigns a weight to the class. The class will receive the assigned percentage of interface bandwidth if there are no strict-priority queues. If there are strict-priority queues, however, the strict-priority queues receive their share of the bandwidth first. The remaining bandwidth is shared in a weighted manner among the class configured with a bandwidth percent. For example, if strict-priority queues take 90 percent of the bandwidth, and you configure 75 percent for a class, the class will receive 75 percent of the remaining 10 percent of the bandwidth.</p> <p><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-fcoe.</p>
<b>Step 8</b>	(Optional) switch(config-pmap-c-que)# <b>no bandwidth percent percentage</b>	Removes the bandwidth specification from this class.

	Command or Action	Purpose
<b>Step 9</b>	(Optional) switch(config-pmap-c-que)# <b>priority level</b> <i>level</i>	Specifies the strict priority levels for the Cisco Nexus 3100 platform switches. These levels can be 1, 2, or 3.  <b>Note</b> The Cisco Nexus 34180YC switch supports a maximum of seven priority queues.
<b>Step 10</b>	(Optional) switch(config-pmap-c-que)# <b>queue-limit</b> <i>queue size</i> [ <b>dynamic</b> <i>dynamic threshold</i> ]	Specifies either the static or dynamic shared limit available to the queue for Cisco Nexus 3100 Series switches. The static queue limit defines the fixed size to which the queue can grow.  The dynamic queue limit allows the queue's threshold size to be decided depending on the number of free cells available, in terms of the alpha value.  <b>Note</b> The queue-limit CLI under queuing policy is enhanced to support 0 queue size. Configuring queue-limit as 0 bytes drops all the packets on the queue.  <b>Note</b> The queue-limit range for the Cisco Nexus 34180YC switch is from 64 to 12582912 bytes.

### Example

This example shows how to define a type queuing policy map for Cisco Nexus 3000 series switches:

```
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)# priority
switch(config-pmap-c-que)# shape kbps 30000000 min 18000000
switch(config-pmap-c-que)# bandwidth percent 20
```

This example shows how to define a type queuing policy map for Cisco Nexus 3100 platform switches:

```
switch# configure terminal
switch(config)# policy-map type queuing p1
switch(config-pmap-que)# class type queuing q3
switch(config-pmap-c-que)# priority level 2
switch(config-pmap-c-que)# shape kbps 30000000 min 18000000
switch(config-pmap-c-que)# class type queuing q2
switch(config-pmap-c-que)# priority level 3
switch(config-pmap-c-que)# class type queuing q1
switch(config-pmap-c-que)# bandwidth percent 30
switch(config-pmap-c-que)# class type queuing q4
switch(config-pmap-c-que)# bandwidth percent 10
```

```

switch(config-pmap-c-que) # class type queuing q5
switch(config-pmap-c-que) # bandwidth percent 10
switch(config-pmap-c-que) # class type queuing q6
switch(config-pmap-c-que) # priority level 1
switch(config-pmap-c-que) # class type queuing q7
switch(config-pmap-c-que) # bandwidth percent 10
switch(config-pmap-c-que) # class type queuing class-default
switch(config-pmap-c-que) # queue-limit dynamic 4
switch(config-pmap-c-que) # bandwidth percent 0

```

This example shows how to define a type queuing policy map for Cisco Nexus 3500 platform switches:

```

switch# configure terminal
switch(config)# policy-map type queuing p1
switch(config-pmap-que) # class type queuing q1
switch(config-pmap-c-que) # bandwidth percent 10
switch(config-pmap-c-que) # priority
switch(config-pmap-c-que) # class type queuing q2
switch(config-pmap-c-que) # bandwidth percent 20
switch(config-pmap-c-que) # class type queuing q3
switch(config-pmap-c-que) # bandwidth percent 30
switch(config-pmap-c-que) # class type queuing q4
switch(config-pmap-c-que) # bandwidth percent 40
switch(config-pmap-c-que) # class type queuing class-default
switch(config-pmap-c-que) # bandwidth percent 10

```

## Configuring an ECN Threshold

You can configure an explicit congestion notification (ECN) threshold per class in a queuing policy and apply it to an interface.

Prior to Cisco NX-OS Release 5.0(3)U4(1), WRED and an ECN can only be enabled or disabled on QoS class in the networkqos policy (with static thresholds). Starting with Cisco NX-OS Release 5.0(3)U4(1), an enhanced ECN marking is supported as follows:



### Note

This feature is not supported on Cisco Nexus 3500 platform switches.

- WRED and ECN thresholds can be configured corresponding to a class from the queuing policy by using the following Steps 1 through 8.



### Note

A WRED and ECN still need to be enabled by the network-qos policy class configuration mode. Not applicable for Cisco Nexus 34180YC switches.

- WRED and ECN can be enabled on a global basis outside the MQC command line. You can configure WRED and an ECN at a global buffer level where you enable WRED and an ECN and specify a threshold at the system level by using the following Steps 1 through 9. If this threshold is exceeded, WRED and ECN are applied on all WRED/ECN enabled classes in the system.
- By default, when WRED and an ECN are enabled, the marking or drop happens based on the class or queue threshold. However, when the global based WRED and ECN is also enabled, by using the **congestion-control random-detect global-buffer** and **wred-queue qos-group-map queue-only** commands, the WRED and ECN marking behavior initiates when either of the class thresholds or global threshold is exceeded.

**Before you begin**

Ensure that you have enabled ECN or WRED on the QoS group that you want in the network-qos policy.

**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 queuing</b> <i>class-map name</i>	Creates or accesses a named object that represents the specified class of traffic in queuing mode. 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-que)# <b>match qos-group</b> <i>qos-group-number</i>	Associates a QoS group to the queuing class map.
<b>Step 4</b>	switch(config-cmap-que)# <b>exit</b>	Exits class mode.
<b>Step 5</b>	switch(config)# <b>policy-map type queuing</b> <i>policy-map name</i>	Creates a named object that represents a set of policies that are to be applied to a set of traffic classes in queuing mode. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
<b>Step 6</b>	switch(config-pmap-que)# <b>class type queuing</b> <i>class-map name</i>	Associates a queuing class map with the policy map, and enters configuration mode for the specified system class.
<b>Step 7</b>	switch(config-pmap-c-que)# <b>random-detect</b> <b>minimum-threshold</b> { <i>min-threshold</i> [bytes   kilobytes   megabytes   packets] } <b>maximum-threshold</b> { <i>max-threshold</i> [bytes   kilobytes   megabytes   packets] } <b>drop-probability</b> <i>drop probability weight weight cap-average</i>	<p>Configures WRED or ECN based on whether the network-qos policy has ECN or WRED enabled. You can specify minimum and maximum thresholds used to drop (WRED) or mark (ECN) packets from the queue. You can configure thresholds by the number of packets, number of bytes, and packets where 1 packet maps to 208 bytes. The minimum and maximum thresholds must be of the same type. If no aggregate arguments are supplied, no aggregate WRED is configured. The default threshold is in packets. The thresholds are from 1 to 83886080.</p> <p>You can also specify the following optional parameters:</p> <ul style="list-style-type: none"> <li>• drop probability—This option specifies the drop probability when the average queue size is between the minimum queue length and maximum queue size. When</li> </ul>

	Command or Action	Purpose
		<p>the queue size is greater than the maximum queue size, 100 percent of frames are dropped. The default drop probability value is 100.</p> <ul style="list-style-type: none"> <li>• <b>weight</b>—This option is used to derive the actual queue size from the current queue size. For example, a weight of 0 means that the actual queue size is equal to the current queue size. The default weight value is 0.</li> <li>• <b>cap-average</b>—This option is used to replace the average queue size with the current queue size if the average queue size is greater than the current queue size.</li> </ul> <p><b>Note</b> N3K-C34180YC does not support cap-average.</p>
<b>Step 8</b>	switch(config-cmap-que)# <b>exit</b>	Exits policy mode.
<b>Step 9</b>	switch(config)# <b>congestion-control random-detect global-buffer minimum-threshold</b> { <i>min-threshold</i> [bytes   kilobytes   megabytes   packets]} <b>maximum-threshold</b> { <i>max-threshold</i> [bytes   kilobytes   megabytes packets]}	<p>Configures the global ECN threshold. You can specify minimum and maximum global buffer thresholds. If the global buffer exceeds these thresholds, the packets in individual queues, which are ECN or WRED enabled, are marked or dropped even if the queue thresholds have not been exceeded. You can configure thresholds by the number of packets, number of bytes, and packets where 1 packet maps to 208 bytes. The minimum and maximum thresholds must be of the same type. If no aggregate arguments are supplied, no aggregate WRED is configured. The default threshold is in packets. The thresholds are from 1 to 83886080.</p> <p><b>Note</b> The Cisco Nexus 34180YC switch does not support <b>congestion-control random-detect global-buffer minimum-threshold</b>.</p>
<b>Step 10</b>	(Optional) switch(config-pmap-nq)# <b>wred-queue qos-group-map queue-only queue-group</b>	Enables ECN marking for the specified QoS group that is based only on a class threshold and independent of the global buffer threshold configuration.
<b>Step 11</b>	(Optional) switch(config-pmap-nq)# <b>show wred-queue qos-group-map</b>	Displays the configuration for the queue-only QoS group maps.

	Command or Action	Purpose
<b>Step 12</b>	(Optional) switch# <b>copy running-config startup-config</b>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

### Example

The following example shows how to configure an ECN threshold per class:

```
switch# configure terminal
switch(config)# class-map type queuing cque_ecn
switch(config-cmap-que)# match qos-group 1
switch(config-cmap-que)# exit
switch(config)# policy-map type queuing pque_ecn
switch(config-pmap-que)# class type queuing cque_ecn
switch(config-pmap-c-que)# random-detect minimum-threshold 20 kilobytes
maximum-threshold 60 kilobytes drop-probability 70 weight 11 cap-average
```

The following example shows how to configure a global ECN threshold in packets, bytes, kilobytes, and megabytes:

```
switch(config)# congestion-control random-detect global-buffer
minimum-threshold 1000 bytes maximum-threshold 1000 bytes
switch(config)#
```

## Configuring Pause Buffer Thresholds and Priority Groups

The ingress queuing policy is used to configure pause buffer thresholds and priority groups. This feature is not supported on Cisco Nexus 3500 Series switches.

### 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-map name</i>	Enters policy-map queuing class mode and identifies the policy map assigned to the type queuing policy map.
<b>Step 3</b>	switch(config-pmap-que)# <b>class type queuing</b> <i>class-map name</i>	References an existing queuing class map in a policy map and enters class mode.  <b>Important</b> This must be defined as a no-drop class in the network-qos policy applied in the system qos.

	Command or Action	Purpose
<b>Step 4</b>	<pre>switch(config-pmap-c-queue)# <b>pause buffer-size</b> buffer-size <b>pause-threshold</b> xoff-size <b>resume-threshold</b> xon-size</pre>	<p>Specifies the buffer threshold settings for pause and resume.</p> <p><b>Note</b> The values mentioned next to each parameter are the recommended values. These values are assigned by the system and will be the default values if no value is defined in the input queuing policy. You can use different values based on your requirement.</p> <p>On 10-Gigabit ports, if the no-drop class has an MTU ≤ 2240:</p> <ul style="list-style-type: none"> <li>• buffer-size—27,456 bytes</li> <li>• pause threshold—12,480 bytes</li> <li>• resume-threshold—0 bytes</li> </ul> <p>On 10-Gigabit ports, if the no-drop class has an MTU &gt; 2240:</p> <ul style="list-style-type: none"> <li>• buffer-size—89,440 bytes</li> <li>• pause threshold—34,320 bytes</li> <li>• resume-threshold—21,840 bytes</li> </ul> <p>On 40-Gigabit ports, if the no-drop class has an MTU of MTU ≤ 2240:</p> <ul style="list-style-type: none"> <li>• buffer-size—83,616 bytes</li> <li>• pause threshold—40,352 bytes</li> <li>• resume-threshold—19,552 bytes</li> </ul> <p>On 40-Gigabit ports, if the no-drop class has an MTU &gt; 2240:</p> <ul style="list-style-type: none"> <li>• buffer-size—1,58,080 bytes</li> <li>• pause threshold— 77,376 bytes</li> <li>• resume-threshold—56,576 bytes</li> </ul>
<b>Step 5</b>	<pre>switch(config-pmap-c-queue)# <b>no pause</b> <b>buffer-size</b> buffer-size <b>pause-threshold</b> xoff-size <b>resume-threshold</b> xon-size</pre>	Removes the buffer threshold settings for pause and resume.
<b>Step 6</b>	<pre>switch(config-pmap-c-queue)# <b>pause</b> <b>priority-group</b> priority group number</pre>	Maps the no-drop class traffic to the priority group specified.
<b>Step 7</b>	<pre>(Optional) switch(config-pmap-c-queue)# <b>queue-limit</b> queue size [<b>dynamic</b> dynamic threshold]</pre>	Specifies either the static or dynamic shared limit available to the queue for Cisco Nexus 3100 Series switches. The static queue limit defines the fixed size to which the queue can grow. The dynamic queue limit allows the queue's threshold size to be decided depending

	Command or Action	Purpose
		on the number of free cells available, in terms of the alpha value.

### Example

This example shows how to configure priority groups for no-drop classes:

```
switch# configure terminal
switch(config-pmap-que)# policy-map type queuing pl
switch(config-pmap-que)# class type queuing cl
switch(config-pmap-c-que)# pause buffer-size 39936 pause-threshold 24960 resume-threshold 12480
switch(config-pmap-c-que)# pause priority-group 1
```

## Information About Marking

Marking is a method that you use to modify the QoS fields of the incoming and outgoing packets.

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</b> [type <b>network-qos</b> ] <i>policy-map name</i>	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</b> [type <b>network-qos</b> ] { <i>class-map name</i>   <b>class-default</b> }	Creates a reference to the <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.



	Command or Action	Purpose
<b>Step 4</b>	switch(config-pmap-c-nq) # <b>set cos</b> <i>cos-value</i>	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 supported only 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** For Cisco Nexus 34180YC switches, the numeric values range from 0 to 63



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

**Table 3: 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	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

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

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	<b>policy-map type qos</b> <i>qos-policy-map-name</i>	Creates or accesses the policy map named qos-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>	<b>class</b> [ <b>type qos</b> ] <i>{class-map-name   class-default}</i>	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>	<b>set dscp</b> <i>dscp-value</i>	Sets the DSCP value to dscp-value. See the Standards DSCP Values table.

### Example

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 cannot set both values because they modify the same field in the IP packet.

**Table 4: 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)
network	Network control precedence (7)
priority	Priority precedence (1)
routine	Routine precedence (0)

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>config terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config) # <b>policy-map</b> [type qos] <i>qos-policy-map-name</i>	Creates or accesses the policy map named <i>policy-map-name</i> , 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>	switch(config-pmap-nq) # <b>class</b> [type qos] { <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 4</b>	switch(config-pmap-c-nq) # <b>set precedence</b> <i>precedence-value</i>	Sets the IP precedence value to precedence-value. You can enter one of the values shown in the Precedence Values table.

### Example

This example shows how to set the precedence marking to 5:

```

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

### Configuring Layer 3 Multicast Queuing

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 global configuration mode.

#### Example

This example shows how to configure a Layer 3 interface:

### Configuring a Service Policy for a Layer 3 Interface

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

#### 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>interface ethernet</b> <i>slot/port</i>	Enters the 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</b> [ <b>type</b> { <b>qos</b>   <b>input</b>   <b>queuing</b> [ <b>input</b>   <b>output</b> ]}] <i>policy-name</i>	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>

	Command or Action	Purpose
		<b>Note</b> The <b>output</b> keyword specifies that this policy map should be applied to traffic transmitted from an interface and the <b>input</b> keyword specifies that this policy map should be applied to traffic transmitted to an interface. You can apply both <b>output</b> and <b>input</b> to a queuing policy. The policy applied at input should have buffer configurations and the policy applied at output should have scheduling and queuing configurations.

### Example

The following 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)#
```

The following example shows how to attach an input qos 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 qos input my_input_qos_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. The **wrr unicast-bandwidth** command configures this bandwidth percentage globally even if you use it on a specific interface.



**Note** The Cisco Nexus 34180YC switch does not support the **wrr unicast-bandwidth** command.

### 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>interface ethernet slot/port</b>	Enters configuration mode for the specified interface.

	Command or Action	Purpose
<b>Step 3</b>	switch(config-if)# <b>wrr unicast-bandwidth</b> percentage-value	Changes the bandwidth allocated to unicast and multicast traffic on traffic congestion globally. The bandwidth-value percentage ranges from 0 to 100 percent.

### Example

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>qos input</b>   <b>queuing</b> [ <b>input</b>   <b>output</b> ]} <i>policy-name</i>	<p>Specifies the policy map to use as the service policy for the system. There are three policy-map configuration modes:</p> <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>

### Example

## 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 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:
<b>Step 4</b>	switch(config-sys-qos)# <b>no service-policy type network-qos</b> <i>policy-map name</i>	Resets the network-wide policy map.
<b>Step 5</b>	switch(config-sys-qos)# <b>no service-policy type queuing [input   output]</b> <i>policy-map name</i>	Resets the output queuing mode policy map.

### Example

## Enabling the Jumbo MTU

You can enable the jumbo Maximum Transmission Unit (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).

When you configure jumbo MTU on a port-channel subinterface you must first enable MTU 9216 on the base interface and then configure it again on the subinterface. If you enable the jumbo MTU on the subinterface before you enable it on the base interface then the following error will be displayed on the console:

```
switch(config)# int po 502.4
switch(config-subif)# mtu 9216
ERROR: Incompatible MTU values
```



### Note

The Cisco Nexus device supports 1 MTU for all classes for all ports.

To use FCoE on switch, add class-fcoe in the custom network-qos policy. If already using FCoE, make sure to add the below lines in the config so that the FCoE does not go down on the switch after enabling the jumbo qos policy.

```
switch# conf t
switch(config)# policy-map type network-qos jumbo
```

```
switch(config-pmap-nqos) # class type network-qos class-fcoe
switch(config-pmap-nqos-c) # end
```

This example shows how to change qos to enable the jumbo MTU:

```
switch# conf t
switch(config) # policy-map type network-qos jumbo
switch(config-pmap-nqos) # class type network-qos class-default
switch(config-pmap-c-nqos) # mtu 9216
```


**Note**

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

## Verifying the Jumbo MTU

On the Cisco Nexus device, traffic is classified into one of eight QoS groups. The MTU is configured at the QoS group level. By default, all Ethernet traffic is in QoS group 0. Use the **show queuing interface ethernet slot/chassis\_number** command to display all qos-groups, control qos group, span qos-group, and pfc statistics.

Use the **show policy-map system type network-qos** command to display the configured HW MTU for the qos-groups.

```
switch(config) # show policy-map system type network-qos
```

```
Type network-qos policy-maps
=====
policy-map type network-qos pn_system
  class type network-qos cn_1
    match qos-group 1
    mtu 9216
  class type network-qos cn_2
    match qos-group 2
    mtu 9216
  class type network-qos cn_3
    match qos-group 3
    mtu 9216
  class type network-qos cn_4
    match qos-group 4
    mtu 9216
  class type network-qos cn_5
    match qos-group 5
    mtu 9216
  class type network-qos cn_6
    match qos-group 6
    mtu 9216
  class type network-qos cn_7
    match qos-group 7
    mtu 9216
  class type network-qos class-default
    match qos-group 0
    mtu 1500
```

```
switch(config) #
```

This example shows queuing interface for the Cisco Nexus 34180YC switch:

```
switch(config) # show queuing interface spq_shape_wred_8q
policy-map type queuing msft_spq_8q
```



```

class type queuing c-out-8q-q5
  priority level 3
class type queuing c-out-8q-q4
  priority level 4
class type queuing c-out-8q-q7
  priority level 1
class type queuing c-out-8q-q6
  priority level 2
class type queuing c-out-8q-q1
  bandwidth remaining percent 60
  random-detect minimum-threshold 120 kbytes maximum-threshold 120 kbytes drop-probability
1 weight 0 ecn
class type queuing c-out-8q-q3
  priority level 5
  shape min 2500 mbps max 2500 mbps
class type queuing c-out-8q-q2
  priority level 6
  shape min 5000 mbps max 5000 mbps
class type queuing c-out-8q-q-default
  bandwidth remaining percent 40
  random-detect minimum-threshold 120 kbytes maximum-threshold 120 kbytes drop-probability
1 weight 0
switch(config-if)#

switch(config-if)# sh queuing interf eth1/29

slot 1
=====

```

## Egress Queuing for Ethernet1/29 [Interface]

QoS-Group#	Bandwidth%	PrioLevel	Shape		Units	QLimit
			Min	Max		
7	-	1	-	-	-	2 (D)
6	-	2	-	-	-	2 (D)
5	-	3	-	-	-	2 (D)
4	-	4	-	-	-	2 (D)
3	-	5	2500000000	2500000000	bps	2 (D)
2	-	6	5000000000	5000000000	bps	2 (D)
1	60	-	-	-	-	2 (D)
0	40	-	-	-	-	2 (D)
+-----+						
QOS GROUP 0						
+-----+						
		Ingress	Egress			
+-----+						
Tx Pkts			0		2887259853	
Tx Byts			0		1562745132	
Tail Drop Pkts			0		1440300408	
Q Depth Byts			0		126160	
ECN Mark Pkts			0		0	
ECN Mark Byts			0		0	
WRED Drop Pkts			0		8925945	
WRED Drop Byts			0		13388917500	
+-----+						
QOS GROUP 1						
+-----+						
		Ingress	Egress			
+-----+						
Tx Pkts			0		15763298	
Tx Byts			0		2170110520	
Tail Drop Pkts			0		20771625	

Q Depth Byts	0	0
ECN Mark Pkts	0	0
ECN Mark Byts	0	0
WRED Drop Pkts	0	0
WRED Drop Byts	0	0
-----		
QOS GROUP 2		
-----		
	Ingress	Egress
-----		
Tx Pkts	0	29990985
Tx Byts	0	2036804540
Tail Drop Pkts	0	31638183
Q Depth Byts	0	0
ECN Mark Pkts	0	0
ECN Mark Byts	0	0
WRED Drop Pkts	0	0
WRED Drop Byts	0	0
-----		
QOS GROUP 3		
-----		
	Ingress	Egress
-----		
Tx Pkts	0	14911143
Tx Byts	0	891878020
Tail Drop Pkts	0	46351570
Q Depth Byts	0	0
ECN Mark Pkts	0	0
ECN Mark Byts	0	0
WRED Drop Pkts	0	0
WRED Drop Byts	0	0
-----		
QOS GROUP 4		
-----		
	Ingress	Egress
-----		
Tx Pkts	0	0
Tx Byts	0	0
Tail Drop Pkts	0	0
Q Depth Byts	0	0
ECN Mark Pkts	0	0
ECN Mark Byts	0	0
WRED Drop Pkts	0	0
WRED Drop Byts	0	0
-----		
QOS GROUP 5		
-----		
	Ingress	Egress
-----		
Tx Pkts	0	0
Tx Byts	0	0
Tail Drop Pkts	0	0
Q Depth Byts	0	0
ECN Mark Pkts	0	0
ECN Mark Byts	0	0
WRED Drop Pkts	0	0
WRED Drop Byts	0	0
-----		
QOS GROUP 6		
-----		
	Ingress	Egress
-----		
Tx Pkts	0	0
Tx Byts	0	0

Tail Drop Pkts	0	0
Q Depth Byts	0	0
ECN Mark Pkts	0	0
ECN Mark Byts	0	0
WRED Drop Pkts	0	0
WRED Drop Byts	0	0
-----		
QOS GROUP 7		
-----		
	Ingress	Egress
-----		
Tx Pkts	0	248
Tx Byts	0	57692
Tail Drop Pkts	0	0
Q Depth Byts	0	0
ECN Mark Pkts	0	0
ECN Mark Byts	0	0
WRED Drop Pkts	0	0
WRED Drop Byts	0	0

```
switch(config-if)#
```

This example shows how to display all qos-groups, control qos group, span qos-group, and pfc statistics:

```
switch# show queuing interface ethernet1/11
```

```
Egress Queuing for Ethernet1/11 [System]
```

QoS-Group#	Bandwidth%	PrioLevel	Min	Shape Max	Units	QLimit
0	100	-	-	-	-	7 (D)
-----						
QOS GROUP 0						
-----						
	Unicast		OOBFC Unicast		Multicast	
-----						
Tx Pkts	0		0		0	
Tx Byts	0		0		0	
Dropped Pkts	0		0		0	
Dropped Byts	0		0		0	
Q Depth Byts	0		0		0	
-----						
QOS GROUP 1						
-----						
	Unicast		OOBFC Unicast		Multicast	
-----						
Tx Pkts	0		0		0	
Tx Byts	0		0		0	
Dropped Pkts	0		0		0	
Dropped Byts	0		0		0	
Q Depth Byts	0		0		0	
-----						
QOS GROUP 2						
-----						
	Unicast		OOBFC Unicast		Multicast	
-----						
Tx Pkts	0		0		0	
Tx Byts	0		0		0	
Dropped Pkts	0		0		0	
Dropped Byts	0		0		0	
Q Depth Byts	0		0		0	
-----						
QOS GROUP 3						
-----						

	Unicast	OOBFC Unicast	Multicast
Tx Pkts	0	0	0
Tx Byts	0	0	0
Dropped Pkts	0	0	0
Dropped Byts	0	0	0
Q Depth Byts	0	0	0
QOS GROUP 4			
	Unicast	OOBFC Unicast	Multicast
Tx Pkts	0	0	0
Tx Byts	0	0	0
Dropped Pkts	0	0	0
Dropped Byts	0	0	0
Q Depth Byts	0	0	0
QOS GROUP 5			
	Unicast	OOBFC Unicast	Multicast
Tx Pkts	0	0	0
Tx Byts	0	0	0
Dropped Pkts	0	0	0
Dropped Byts	0	0	0
Q Depth Byts	0	0	0
QOS GROUP 6			
	Unicast	OOBFC Unicast	Multicast
Tx Pkts	0	0	0
Tx Byts	0	0	0
Dropped Pkts	0	0	0
Dropped Byts	0	0	0
Q Depth Byts	0	0	0
QOS GROUP 7			
	Unicast	OOBFC Unicast	Multicast
Tx Pkts	0	0	0
Tx Byts	0	0	0
Dropped Pkts	0	0	0
Dropped Byts	0	0	0
Q Depth Byts	0	0	0
CONTROL QOS GROUP			
	Unicast	OOBFC Unicast	Multicast
Tx Pkts	1055	0	0
Tx Byts	83011	0	0
Dropped Pkts	7	0	0
Dropped Byts	508	0	0
Q Depth Byts	0	0	0
SPAN QOS GROUP			
	Unicast	OOBFC Unicast	Multicast
Tx Pkts	0	0	0
Tx Byts	0	0	0

```

| Dropped Pkts |                0|                0|                0|
| Dropped Byts |                0|                0|                0|
| Q Depth Byts |                0|                0|                0|
+-----+

```

#### Port Egress Statistics

```

-----
WRED Drop Pkts                0

```

#### Ingress Queuing for Ethernet1/11

```

-----
QoS-Group#      Buff Size      Pause      Pause Th      Resume Th      QLimit
-----
      7          -          -          -          -      11210992 (S)
      6          -          -          -          -      11210992 (S)
      5          -          -          -          -      11210992 (S)
      4          -          -          -          -      11210992 (S)
      3          -          -          -          -      11210992 (S)
      2          -          -          -          -      11210992 (S)
      1          -          -          -          -      11210992 (S)
      0          -          -          -          -      11210992 (S)

```

#### Port Ingress Statistics

```

-----
Ingress MMU Drop Pkts                0
Ingress MMU Drop Bytes                0

```

#### PFC Statistics

```

-----
TxPPP:                0, RxPPP:                0
-----
COS  QoS  Group      PG   TxPause  TxCount      RxPause  RxCount
-----
  0      -        7  Inactive      0      Inactive      0
  1      -        7  Inactive      0      Inactive      0
  2      -        7  Inactive      0      Inactive      0
  3      -        7  Inactive      0      Inactive      0
  4      -        7  Inactive      0      Inactive      0
  5      -        7  Inactive      0      Inactive      0
  6      -        7  Inactive      0      Inactive      0
  7      -        7  Inactive      0      Inactive      0
-----

```

## 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.



**Note** The Cisco Nexus 34180YC switch does not support **untagged cos** *cos-value*.

**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>interface</b> { <b>ethernet</b> [chassis/]slot/port   <b>port-channel</b> channel-number}	Enters the configuration mode for the specified interface or port channel.
<b>Step 3</b>	switch(config-if)# <b>untagged cos</b> cos-value	Configures the untagged CoS value. Values can be from 1 to 7.

**Example**

The following 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
```

## Configuring an 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 global configuration mode.
<b>Step 2</b>	switch(config)# <b>interface</b> { <b>ethernet</b> [chassis/]slot/port   <b>port-channel</b> channel-number}	Enters the 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 input</b> policy-name	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.

**Example**

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 the QoS Configuration

To verify the QoS configurations, perform one of these tasks:

Command	Purpose
switch# <b>show class-map</b>	Displays the class maps defined on the device.
switch# <b>show policy-map</b> [ <i>name</i> ]	Displays the policy maps defined on the device. 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> { <b>network-qos</b>   <b>qos</b>   <b>queuing</b> } [ <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.  <b>Note</b> This command may not work on Cisco Nexus 3500 Series switches.
switch# <b>show running-config ipqos</b>	Displays information about the running configuration for QoS.
switch# <b>show startup-config ipqos</b>	Displays information about the startup configuration for QoS.
switch# <b>show queuing interface ethernet slot-no/port-no</b>	Displays the queuing information on interfaces.
switch# <b>show queuing</b>	Displays the queuing information configured for all interfaces. It includes shaper configuration information for each class, the control queue Tx and drop statistics for each port, and WRED drop packet counts.

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-nqos)# class type network-qos cnq1
switch(config-pmap-nqos-c)# mtu 2200
switch(config-pmap-nqos-c)# pause no-drop
switch(config-pmap-nqos-c)# set cos 4
switch(config-pmap-nqos-c)# exit
switch(config-pmap-nqos)# class type network-qos cnq6
switch(config-pmap-nqos-c)# mtu 2200
switch(config-pmap-nqos-c)# pause no-drop
switch(config-pmap-nqos-c)# set cos 5
switch(config-pmap-nqos-c)# congestion-control random-detect ecn
switch(config-pmap-nqos-c)# exit
switch(config-pmap-nqos)# class type network-qos class-default
switch(config-pmap-nqos-c)# mtu 9216
switch(config-pmap-nqos-c)# exit
switch(config-pmap-nqos)# system qos
switch(config-sys-qos)# service-policy type network-qos pnqos
switch(config-sys-qos)#

```

This example shows how to configure an output queuing policy:

```

switch(config)# class-map type queuing cqul
switch(config-cmap-que)# match qos-group 1
switch(config-cmap-que)# exit
switch(config)# class-map type queuing cqu6
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 cqul
switch(config-pmap-c-que)# bandwidth percent 10
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# class type queuing cqu6
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 an input queuing policy:

```

switch(config)# class-map type queuing cqul
switch(config-cmap-que)# match qos-group 1
switch(config-cmap-que)# exit
switch(config)# class-map type queuing cqu6
switch(config-cmap-que)# match qos-group 6
switch(config-cmap-que)# exit
switch(config)# policy-map type queuing piqu
switch(config-pmap-que)# class type queuing cqul
switch(config-pmap-c-que)# pause buffer-size 39936 pause-threshold 24960 resume-threshold
12480
switch(config-pmap-c-que)# pause priority-group 1
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# class type queuing cqu6
switch(config-pmap-c-que)# pause priority-group 3
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# class type queuing class-default
switch(config-pmap-c-que)# queue-limit dynamic 2
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# exit

```



```
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing output piqu
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: Tue Dec 10 08:29:08 2013

version 6.0(2)U2(1)
class-map type qos match-all c1
  match cos 1
class-map type qos match-all c2
  match cos 2
.
.
.
class-map type qos match-any cq1
```

```

    match cos 4
class-map type qos match-any cq2
    match cos 5
class-map type qos match-all dscp
    match precedence 0
class-map type qos match-all cq1_1
    match cos 4
    match precedence 7
class-map type qos match-all cq2_1
    match cos 5
    match precedence 3
class-map type qos match-all cMap_Cos_1
    match cos 1
class-map type qos match-all cMap_Cos_2
    match cos 2
.
.
.

class-map type queuing cMap_Qing_match_qGrp_7
    match qos-group 7
policy-map type qos inpq
    class c5
        set qos-group 5
.
.
.

policy-map type queuing piqu
    class type queuing cqul
        pause buffer-size 39936 pause-threshold 24960 resume-threshold 12480
        pause priority-group 1
    class type queuing cqu6
        pause priority-group 3
.
.
.

class-map type network-qos cMap_NQ_match_qGrp_7
    match qos-group 7
policy-map type network-qos pnqos
    class type network-qos cnq1
        mtu 2200
        pause no-drop
        set cos 4
    class type network-qos cnq6
        mtu 2200
        pause no-drop
        set cos 5
        congestion-control random-detect ecn
.
.
.

system qos
    service-policy type qos input pMap_Qos_system
    service-policy type network-qos pMap_NQ_system
    service-policy type queuing output pMap_Qing_system

interface Ethernet1/1/1
    priority-flow-control mode on

interface Ethernet1/32
    priority-flow-control mode on

```

This example shows how to display the QoS groups that are mapped to the egress queue:

```
switch(config)# wrr-queue qos-group-map 3 1
switch(config)# show wrr-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 how to display 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 cqul
  match qos-group 1
```

```
class-map type queuing cqu6
  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 how to display 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 piqu
  class type queuing cqul
    pause buffer-size 39936 pause-threshold 24960 resume-threshold 12480
    pause priority-group 1
  class type queuing cqu6
    pause priority-group 3
  class type queuing class-default
    bandwidth percent 100
    queue-limit dynamic 2

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 how to display 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 2200
    pause no-drop
    set cos 4
  class type network-qos cnq6
    match qos-group 6

    mtu 2200
    pause no-drop
    set cos 5
    congestion-control random-detect ecn
  class type network-qos class-default
    match qos-group 0

    mtu 9216

```

```

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

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 how to display the service policy maps configured on the interfaces:

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

```

Global statistics status :   enabled

Ethernet1/1

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

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:    enabled

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 how to display the queuing information configured for all interfaces:

```

switch# show queuing
Egress Queuing for Ethernet1/1 [Interface]
-----
QoS-Group# Bandwidth% PrioLevel           Min           Shape           Units
                                     Max
-----
      0           10         -           0           0           -
      1           10         -           0           0           -
      2           10         -           0           0           -
      3           10         1           0           0           -
      4           10         -           0           0           -
      5           10         2           0           0           -
      6           10         -           0           0           -
      7           10         -           0           0           -
      9            0         -           0           0           -
+-----+
|                                     | QOS GROUP 0 |
+-----+
|          | Unicast |          | Multicast |          |
+-----+
| Tx Pkts |          | 0 |          | 0 |
| Tx Byts |          | 0 |          | 0 |
| Dropped Pkts |          | 0 |          | 0 |
| Dropped Byts |          | 0 |          | 0 |
+-----+
|                                     | QOS GROUP 1 |
+-----+
|          | Unicast |          | Multicast |          |
+-----+
| Tx Pkts |          | 0 |          | 0 |
| Tx Byts |          | 0 |          | 0 |
| Dropped Pkts |          | 0 |          | 0 |
| Dropped Byts |          | 0 |          | 0 |
+-----+
|                                     | QOS GROUP 2 |
+-----+
|          | Unicast |          | Multicast |          |
+-----+
| Tx Pkts |          | 0 |          | 0 |
| Tx Byts |          | 0 |          | 0 |
| Dropped Pkts |          | 0 |          | 0 |
| Dropped Byts |          | 0 |          | 0 |
+-----+
|                                     | QOS GROUP 3 |
+-----+
|          | Unicast |          | Multicast |          |
+-----+
| Tx Pkts |          | 0 |          | 0 |
| Tx Byts |          | 0 |          | 0 |
| Dropped Pkts |          | 0 |          | 0 |
| Dropped Byts |          | 0 |          | 0 |
+-----+
|                                     | QOS GROUP 4 |
+-----+

```

```

+-----+
|          | Unicast          | Multicast          |
+-----+
| Tx Pkts|          0|          0
| Tx Byts|          0|          0
| Dropped Pkts|          0|          0
| Dropped Byts|          0|          0
+-----+
| QOS GROUP 5 |
+-----+
|          | Unicast          | Multicast          |
+-----+
| Tx Pkts|          0|          0
| Tx Byts|          0|          0
| Dropped Pkts|          0|          0
| Dropped Byts|          0|          0
+-----+
| QOS GROUP 6 |
+-----+
|          | Unicast          | Multicast          |
+-----+
| Tx Pkts|          0|          0
| Tx Byts|          0|          0
| Dropped Pkts|          0|          0
| Dropped Byts|          0|          0
+-----+
| QOS GROUP 7 |
+-----+
|          | Unicast          | Multicast          |
+-----+
| Tx Pkts|          0|          0
| Tx Byts|          0|          0
| Dropped Pkts|          0|          0
| Dropped Byts|          0|          0
+-----+
| CONTROL QOS GROUP 9 |
+-----+
|          | Unicast          | Multicast          |
+-----+
| Tx Pkts|        1901|          0
| Tx Byts|       145235|          0
| Dropped Pkts|          0|          0
| Dropped Byts|          0|          0
+-----+
Port Egress Statistics
+-----+
WRED Drop Pkts          0

```

...

Egress Queuing for Ethernet1/4 [Interface]

```

+-----+
QoS-Group# Bandwidth% PrioLevel          Min          Shape Max          Units
+-----+
0          100          -          0          0          -
9          0          -          0          0          -
+-----+
| QOS GROUP 0 |
+-----+
|          | Unicast          | Multicast          |
+-----+
| Tx Pkts|          0|          0

```

```

|          Tx Byts|          0|          0
|   Dropped Pkts|          0|          0
|   Dropped Byts|          0|          0
+-----+
|                                CONTROL QOS GROUP 9                                |
+-----+
|          | Unicast          | Multicast          |
+-----+
|          Tx Pkts|          8634|          0
|          Tx Byts|        1218248|          0
|   Dropped Pkts|          0|          0
|   Dropped Byts|          0|          0
+-----+
Port Egress Statistics
+-----+
WRED Drop Pkts          0

```

## Monitoring the QoS Packet Buffer

Cisco Nexus 3000 series switches have a 9-MB buffer memory and the Cisco Nexus 3100 platform switches have a 12-MB buffer memory that divides into a dedicated per port and dynamic shared memory. Each front-panel port has eight unicast and four multicast queues in egress. The Cisco Nexus 3100 platform switches have eight unicast and eight multicast queues in egress. In the scenario of burst or congestion, each egress port consumes buffers from the dynamic shared memory.

For the Cisco Nexus 3000 series switches, you can display the real-time status of the shared buffer per port. For the Cisco Nexus 3100 platform switches, you can also display the peak status of the shared buffer per port. 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.





**Note** For the Cisco Nexus 34180YC switch:

- The maximum amount of data in a cell is 80 bytes.
- The total number of cells available is 194224.
- The total memory is 15 MB (194224 \* 80 = 15537920 bytes).

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

```
slot 1
=====
```

```
INSTANCE: 0
```

```
=====
```

	Output Buffer Pool Utilization (in cells)			
	BP-0	BP-1	BP-2	BP-3
Total Instant Usage	1537	0	-	-
Remaining Instant Usage	192687	5200	-	-
Peak/Max Cells Used	38965	228	-	-
Switch Cell Count	194224	5200	-	-

```
switch(config-if)#
```

Starting with Release 6.0(2)U5(1), the buffer is carved into two pools: pool 0 and pool 1. Cisco Nexus 3000 Series switches that have Broadcom T+ have 9MB of total buffer and Broadcom T2 have 12MB of total buffer. There are three buffer modes – all, default, and none. The default mode of buffer in Cisco Nexus 3000 series switches is all.



**Note** The Cisco Nexus 34180YC switch does not support the all, default, and none buffer modes.

In default mode all for T+, there are 10 unicast queues and 5 multicast queues. Each queue in T+ consumes 8 cells. Therefore, 10x8+5x8=120 cells are consumed by each port. T+ has total 46080 cells. The all mode consumes 120 cells/port in T+.

In default mode all for T2, there are 10 unicast queues and 10 multicast queues. Each queue in T2 consumes 11 cells. Therefore, 10x11+10x11=220 cells are consumed by each port. T2 has total 61440 cells. The all mode consumes 220 cells/port in T2.

Cisco Nexus 3000 series switches that are based out of T+ and T2 have 48 CPU multicast queues.

The **default** mode consumes for one default unicast queue, multicast queue, unicast CPU queue (q9), and multicast cpu queue (q9) totaling 11x4=44 cells. The **none** mode consumes only for unicast CPU queue (q9) and multicast CPU queue (q9) – totaling 11x2=22 cells.

The maximum cells consumed in each platform are:

- Platform T2 based Product– Max Cells for All Mode is 23586 cells.

- Platform T2 based Product—Max Cells for Default Mode is 29903 cells.
- Platform T2 based Product—Max Cells for None Mode is 30677 cells.

The **show running | grep hardware** command displays the configured mode in the CLI output:

```
switch# show running | grep hardware
hardware profile portmode 48X10G+breakout6x40g
hardware qos min-buffer qos-group none
```

Usage information of shared buffer resources:

- Total Instant Usage—Current buffer usage in terms of the number of cells on a global basis.
- Remaining Instant Usage—Effective free number of cells available on a global basis.
- Max Cell Usage—Maximum buffer usage that is seen until the system buffer maximum cell usage counter is last cleared.
- 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 8 multicast (Q1-Q8) instant cell usage. The example above shows that the multicast queue Q1 is consuming 3807 cells instantaneously on port 9.

The following example shows how to clear the system buffer maximum cell usage counter:

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

The following 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.



#### Note

The Cisco Nexus 34180YC switch does not support buffer utilization threshold on a per port basis.

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



## CHAPTER 4

# Configuring Priority Flow Control

---

This chapter contains the following sections:

- [About Priority Flow Control, on page 61](#)
- [Guidelines and Limitations for Priority Flow Control, on page 62](#)
- [Default Settings for Priority Flow Control, on page 63](#)
- [Enabling Priority Flow Control on a Traffic Class, on page 64](#)
- [Configuring Priority Flow Control, on page 66](#)
- [Reserving mmu-buffer for PFC, on page 66](#)
- [Configuring a Priority Flow Control Watchdog Interval, on page 67](#)
- [Verifying the Priority Flow Control Configuration, on page 69](#)
- [Monitoring PFC Frame Counter Statistics, on page 69](#)
- [Configuration Examples for Priority Flow Control , on page 70](#)

## About Priority Flow Control

Priority flow control (PFC; IEEE 802.1Qbb), which is also referred to as Class-based Flow Control (CBFC) or Per Priority Pause (PPP), is a mechanism that prevents frame loss that is due to congestion. PFC functions on a per class-of-service (CoS) basis.

When a buffer threshold is exceeded due to congestion, PFC sends a pause frame that indicates which CoS value needs to be paused. A PFC pause frame contains a 2-octet timer value for each CoS that indicates the length of time that the traffic needs to be paused. The unit of time for the timer is specified in pause quanta. A quanta is the time that is required for transmitting 512 bits at the speed of the port. The range is from 0 to 65535. A pause frame with a pause quanta of 0 indicates a resume frame to restart the paused traffic.

By default, PFC is in the auto mode. However, no particular traffic class is enabled for pause.



---

**Note**

Only certain classes of service of traffic can be flow controlled while other classes are allowed to operate normally.

PFC asks the peer to stop sending frames of a particular CoS value by sending a pause frame to a well-known multicast address. This pause frame is a one-hop frame that is not forwarded when received by the peer. When the congestion is mitigated, PFC can request the peer to restart transmitting frames.

---



**Note** RDMA over Converged Ethernet (RoCE) v1 and v2 protocols are supported on Cisco Nexus 3000 Series switches.

## Guidelines and Limitations for Priority Flow Control

PFC has the following guidelines and limitations:

- If PFC is enabled on a port or a port channel, it does not cause a port flap.
- Ensure that ports or port channels have enough resources before enabling PFC on them.
- PFC configuration enables PFC in both the send (Tx) and receive (Rx) direction.
- Only an exact match of the no-drop CoS is considered as a successful negotiation of PFC by the Data Center Bridging Exchange Protocol (DCBXP).
- Configuration time quanta of the pause frames is not supported.
- The configuration does not support pausing selected streams that are mapped to a particular traffic-class queue. All flows that are mapped to the class are treated as no-drop. It blocks out scheduling for the entire queue, which pauses traffic for all the streams in the queue. To achieve lossless service for a no-drop class, we highly recommend that you have only the no-drop class traffic on the queue.
- For VLAN-tagged packets, priority is always assigned based on the 802.1p field in the VLAN tag and takes precedence over the assigned internal priority(qos-group). DSCP or IP access-list classification cannot be performed on VLAN-tagged frames
- When a no-drop class is classified based on 802.1p CoS x and assigned an internal priority value (qos-group) of y, we recommend that you use the internal priority value x to classify traffic on 802.1p CoS only, and not on any other field. For x, the packet priority assigned is x if the classification is not based on CoS, which results in packets of the internal priority that is x and y to map to the same priority x.
- The PFC feature supports up to three no-drop classes of any MTU size. However, there is a limit on the number of PFC-enabled interfaces based on the following factors:
  - MTU size of the no-drop class
  - Number of 10G and 40G ports
  - Pause buffer size configuration in the input queuing policies
- Interface QoS policy takes precedence over the system policy. PFC priority derivation also occurs in the same order.
- Ensure that you apply the same interface-level QoS policy on all PFC-enabled interfaces for both ingress and egress.

**Caution**

Irrespective of the PFC configuration, we recommend that you stop traffic before you apply or remove the queuing policy that has strict priority levels at the interface level or the system level.

- To achieve end-to-end lossless service over the network, we recommend that you enable PFC on each interface through which the no-drop class traffic flows #(Tx/Rx).
- To achieve lossless service for a no-drop class, it is recommended that you maintain only no-drop class traffic on the egress queue.
- We recommend that you change the PFC configuration when there is no traffic; otherwise, packets already in the memory management unit (MMU) of the system might not get the expected treatment.
- The buffers required for PFC are best allocated automatically. However, you can change the buffer thresholds by configuring input queuing policies.
- For no-drop classes classified based on DSCP/IP access-lists (non CoS based classifications), we highly recommend that you use the same qos-group value as the match CoS value.
- Do not enable WRED on a no-drop class because it results in egress queue drops.
- When you configure a port from the 40 Gigabit Ethernet mode to the 10 Gigabit Ethernet mode or from the 10 Gigabit Ethernet mode to the 40 Gigabit Ethernet mode, the affected ports will be administratively unavailable and PFC will be disabled on these ports. To make these ports available, use the **no shut** command. After the ports are available, PFC will become enabled on them.
- Beginning with Cisco NX-OS Release 7.0(3)I4(2), the **no lldp tlv-select dcbxp** command is enhanced so that the PFC is disabled for interfaces on both sides of back-to-back switches.
- When the 'hardware qos pfc mc-drop' configuration is enabled on Cisco Nexus 3064 switches, and if the no-drop and drop qos-groups are mapped to the same queue, then flapping the link or removing/adding PFC configurations will result in the drop-multicast feature not working correctly. This is due to the hardware limitation of having only four multicast queues in Cisco Nexus 3064 switches. To avoid this issue, do one of the following:
  - Specify both of the qos-groups that correspond to a single multicast queue as no-drop.
  - Change the mapping of the multicast queue to qos-groups using the **wrr-queue qos-group-map <queue-no> <qos-groups that are no-drop>** command.

## Default Settings for Priority Flow Control

The following table lists the default setting for PFC.

**Table 5: Default PFC Setting**

Parameter	Default
PFC	Auto

# Enabling Priority Flow Control on a Traffic Class

You can enable PFC on a particular traffic class:

## 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 cos</b> <i>cos-value</i>	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>exit</b>	Exits class-map mode and enters global configuration mode.
<b>Step 5</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 6</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 7</b>	switch(config-pmap-c-qos)# <b>set qos-group</b> <i>qos-group-value</i>	Configures one or more qos-group values to match on for classification of traffic into this class map. There is no default value.
<b>Step 8</b>	switch(config-pmap-c-qos)# <b>exit</b>	Exits policy-map mode and enters global configuration mode.
<b>Step 9</b>	switch(config)# <b>interface type slot/port</b>	Enters the configuration mode for the specified interface.
<b>Step 10</b>	switch(config-if)# <b>service-policy type qos</b> <i>input policy-name</i>	Applies the policy map of type qos to the specific interface.

	Command or Action	Purpose
<b>Step 11</b>	switch(config-if)# <b>exit</b>	Exits interface configuration mode and enters global configuration mode.
<b>Step 12</b>	switch(config)# <b>class-map type network-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 13</b>	switch(config-cmap-nq)# <b>match qos-group</b> <i>qos-group-value</i>	Configures the traffic class by matching packets based on a list of QoS group values. Values can range from 0 to 7. QoS group 0 is equivalent to class-default.
<b>Step 14</b>	switch(config-cmap-nq)# <b>exit</b>	Exits class-map mode and enters global configuration mode.
<b>Step 15</b>	switch(config)# <b>policy-map type network-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 16</b>	switch(config-pmap-nqos)# <b>class type</b> <b>network-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 17</b>	switch(config-pmap-c-nq)# <b>pause no-drop</b>	Configures a no-drop class.
<b>Step 18</b>	switch(config-pmap-c-nq)# <b>exit</b>	Exits policy-map mode and enters global configuration mode.
<b>Step 19</b>	switch(config)# <b>system qos</b>	Enters system class configuration mode.
<b>Step 20</b>	switch(config-sys-qos)# <b>service-policy type</b> <b>network-qos</b> <i>policy-name</i>	Applies the policy map of type network-qos at the system level or to the specific interface.

### Example

This example shows how to enable PFC on a traffic class .

```
switch# configure terminal
switch(config)# class-map type qos c1
switch(config-cmap-qos)# match cos 3
switch(config-cmap-qos)# exit
switch(config)# policy-map type qos p1
switch(config-pmap-qos)# class type qos c1
```

```

switch(config-pmap-c-qos) # set qos-group 3
switch(config-pmap-c-qos) # exit
switch(config) # interface ethernet 1/1
switch(config-if) # service-policy type qos input p1
switch(config-if) # exit
switch(config) # class-map type network-qos c1
switch(config-cmap-nq) # match qos-group 3
switch(config-cmap-nq) # exit
switch(config) # policy-map type network-qos p1
switch(config-pmap-nqos) # class type network-qos c1
switch(config-pmap-nqos-c) # pause no-drop
switch(config-pmap-nqos-c) # exit
switch(config) # system qos
switch(config-sys-qos) # service-policy type network-qos p1

```

## Configuring Priority Flow Control

You can configure PFC on a per-port basis to enable the no-drop behavior for the CoS as defined by the active network qos policy. PFC can be configured in one of these three modes:

- **auto**—Enables the no-drop CoS values to be advertised by the DCBXP and negotiated with the peer. A successful negotiation enables PFC on the no-drop CoS. Any failures because of a mismatch in the capability of peers causes the PFC not to be enabled.
- **on**—Enables PFC on the local port regardless of the capability of the peers.
- **off**—Disables PFC on the local port.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	<b>interface ethernet</b> [ <i>slot/port-number</i> ]	Enter the interface mode on the specified interface.
<b>Step 3</b>	<b>priority-flow-control mode</b> { <b>auto</b>   <b>off</b>   <b>on</b> } <b>priority-flow-control mode</b> { <b>auto</b>   <b>off</b>   <b>on</b> }	Sets the PFC to the auto, off, or on mode. By default, PFC mode is set to auto on all ports.
<b>Step 4</b>	<b>exit</b>	Exits interface configuration mode.
<b>Step 5</b>	<b>show interface priority-flow-control</b>	Displays the status of PFC on all interfaces.

## Reserving mmu-buffer for PFC

Complete the following steps to reserve mmu-buffer for PFC.



**Procedure**

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	<b>hardware profile pfc mmu buffer-reservation ?</b>  <b>Example:</b> switch(config)# hardware profile pfc mmu buffer-reservation ?	Reserves the mmu-buffer for PFC.  <0-100> Percentage of shared pool buffers to be reserved

## Configuring a Priority Flow Control Watchdog Interval

You can configure a PFC watchdog interval to detect whether packets in a no-drop queue are being drained within a specified time period. When the time period is exceeded, all incoming and outgoing packets are dropped on interfaces that match the PFC queue that is not being drained. This feature is supported beginning with Cisco NX-OS Release 6.0(3)U6(9) and only for Cisco Nexus 3000 Series switches.

Starting with Cisco NX-OS Release 7.0(3)I6(1), ingress packets will be dropped for matching to shutdown PFC queue or qos-group of a physical port, and the show command displays the status.

**Procedure**

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> switch# configure terminal switch(config)#	Enters global configuration mode.
<b>Step 2</b>	<b>priority-flow-control auto-restore multiplier</b> <i>value</i>	Configures a value for the PFC auto-restore multiplier.
<b>Step 3</b>	<b>priority-flow-control fixed-restore multiplier</b> <i>value</i>	Configures a value for the PFC fixed-restore multiplier.
<b>Step 4</b>	<b>priority-flow-control watch-dog-interval {on   off}</b>  <b>Example:</b> switch(config)# priority-flow-control watch-dog-interval on	Globally enables or disables the PFC watchdog interval for all interfaces.  <b>Note</b> You can use this same command in interface configuration mode to enable or disable the PFC watchdog interval for a specific interface.
<b>Step 5</b>	<b>priority-flow-control watch-dog interval</b> <i>value</i>  <b>Example:</b> switch(config)# priority-flow-control watch-dog interval 200	Specifies the watchdog interval value. The range is from 100 to 1000 milliseconds.

	Command or Action	Purpose
<b>Step 6</b>	<b>priority-flow-control watch-dog shutdown-multiplier <i>multiplier</i></b>  <b>Example:</b> <pre>switch(config)# priority-flow-control watch-dog shutdown-multiplier 5</pre>	Specifies when to declare the PFC queue as stuck. The range is from 1 to 10, and the default value is 1.
<b>Step 7</b>	<b>(Optional) sh queuing pfc-queue [interface] [ethernet] [detail]</b>  <b>Example:</b> <pre>switch(config)# sh queuing pfc-queue interface ethernet 1/1 detail</pre>	<p>Displays the PFCWD statistics. Starting with Cisco NX-OS Release 7.0(3)I6(1), you can use the detail option to account for Ingress drops.</p> <pre>  QOS GROUP 2 [Active] PFC [YES] PFC-COS [1] +-----+     Stats   +-----+     Shutdown    0    Restored    0    Total bytes drained    0    Total pkts dropped    0    Aggregate pkts dropped    0    Ingress Pkts dropped    0    Aggregate Ingress Pkts dropped    0  Global watch-dog interval [Enabled] +-----+ +-----+ Global PFC watchdog configuration details  PFC watch-dog poll interval : 200 ms PFC watch-dog shutdown multiplier : 5 PFC watch-dog auto-restore multiplier : 10 PFC watch-dog fixed-restore multiplier : 0 PFC watchdog internal-interface multiplier : 0</pre>
<b>Step 8</b>	<b>(Optional) clear queuing pfc-queue [interface] [ethernet] [intf-name]</b>  <b>Example:</b> <pre>switch(config)# clear queuing pfc-queue interface ethernet 1/1</pre>	Clears the environment variable PFCWD statistics.

	Command or Action	Purpose
<b>Step 9</b>	(Optional) <b>priority-flow-control recover interface [ethernet] [intf-name] [qos-group &lt;0-7&gt;]</b>  <b>Example:</b>  switch# priority-flow-control recover interface ethernet 1/1 qos-group 3	Recovers the interface manually.

## Verifying the Priority Flow Control Configuration

To display the PFC configuration, perform the following task:

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>	Enter global configuration mode.
<b>Step 2</b>	<b>show interface priority-flow-control</b>	Displays the status of PFC on all interfaces.
<b>Step 3</b>	(Optional) <b>show interface priority-flow-control detail</b>	Displays the status of PFC for each priority level for each interface.

## Monitoring PFC Frame Counter Statistics

You can monitor the Tx and Rx counters for PFC-enabled devices either at an interface level or at a per priority (CoS) level for each interface.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>show int priority-flow-control [detail]</b>	

### Example

This example shows how to display PFC frame counter statistics for each priority level for each interface:

```
switch# show int priority-flow-control detail
```

```
Ethernet1/1/1:
  Admin Mode: On
  Oper Mode: On
  VL bitmap: (14)
  Total Rx PFC Frames: 0
  Total Tx PFC Frames: 0
  -----
```

```

-----
      | Priority0 | Priority1 | Priority2 | Priority3 | Priority4 |
Priority5 | Priority6 | Priority7 |
-----
      Rx  | 0      | 0      | 0      | 0      | 0      |
          | 0      | 0      |
-----
      Tx  | 0      | 0      | 0      | 0      | 0      |
          | 0      | 0      |
-----
Ethernet1/1/2:
  Admin Mode: Auto
  Oper Mode: Off
  VL bitmap:
  Total Rx PFC Frames: 0
  Total Tx PFC Frames: 0
-----
      | Priority0 | Priority1 | Priority2 | Priority3 | Priority4 |
Priority5 | Priority6 | Priority7 |
-----
      Rx  | 0      | 0      | 0      | 0      | 0      |
          | 0      | 0      |
-----
      Tx  | 0      | 0      | 0      | 0      | 0      |
          | 0      | 0      |

```

This example shows how to display PFC frame counter statistics for each interface:

```

switch# show int priority-flow-control
=====
Port                Mode Oper (VL bmap)  RxPPP  TxPPP
=====
Ethernet1/1/1       On  On  (14)          0       0
Ethernet1/1/2       Auto Off           0       0
Ethernet1/1/3       Auto On  (14)          0       0
Ethernet1/15        Auto On  (14)          0       0
Ethernet1/15        Auto On  (14)          0       0
Ethernet1/15        Auto On  (14)          0       0
Ethernet1/15        Auto On  (14)          0       0
Ethernet1/25        Auto On  (14)          0       0
Ethernet1/32        On  On  (14)          0       0
switch#

```

## Configuration Examples for Priority Flow Control

The following example shows how to configure PFC.

```

switch# configure terminal
switch(config)# interface ethernet 5/5
switch(config-if)# priority-flow-control mode on

```



## CHAPTER 5

# Configuring Traffic Shaping

This chapter contains the following sections:

- [About Traffic Shaping, on page 71](#)
- [Guidelines and Limitations for Traffic Shaping, on page 72](#)
- [Configuring Traffic Shaping, on page 72](#)
- [Verifying Traffic Shaping, on page 73](#)
- [Configuration Example for Traffic Shaping, on page 73](#)

## About Traffic Shaping

Traffic shaping allows you to control the traffic going out an interface in order to match its flow to the speed of the remote target interface and to ensure that the traffic conforms to policies contracted for it. Traffic that is adhering to a particular profile can be shaped to meet downstream requirements, which eliminates bottlenecks in topologies with data-rate mismatches.

Traffic shaping regulates and smooths out the packet flow by imposing a maximum traffic rate for each port's egress queue. Packets that exceed the threshold are placed in the queue and are transmitted later. This process is similar to traffic policing; however, the packets are not dropped. Because packets are buffered, traffic shaping minimizes packet loss (based on the queue length), which provides a better traffic behavior for TCP traffic.

Using traffic shaping, you can control access to available bandwidth, ensure that traffic conforms to the policies established for it, and regulate the flow of traffic in order to avoid congestion that can occur when the egress traffic exceeds the access speed of its remote, target interface. For example, you can control access to the bandwidth when policy dictates that the rate of a given interface should not, on average, exceed a certain rate even though the access rate exceeds the speed.

The traffic shaping rate can be configured in kilobits per second (kbps) or packets per second (PPS) and is applied to unicast queues. Queue length thresholds are configured using weighted random early detection (WRED) configuration.

Traffic shaping can be configured at the system level or the interface level. System level queuing policies can be overridden by interface queuing policies.

# Guidelines and Limitations for Traffic Shaping

- Traffic shaping might increase the latency of packets due to queuing, because it falls back to store-and-forward mode when packets get queued.
- For traffic shaping to function properly, the store and forwarding switching mode needs to be enabled.

## Configuring Traffic Shaping

You can configure a maximum traffic rate to regulate traffic flow.

### Before you begin

Configure random-detect minimum and maximum thresholds for packets.

Configure congestion control random detection on the network QoS class map by using the **congestion-control random detect** command under the network-qos class-map.

Both QoS and network QoS policies must be applied for queuing to work. This prerequisite exists for configuring any queuing policy.

### 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-que)# <b>shape {kbps   mbps   gbps} burst size min minimum bandwidth</b>	Specifies the burst size and minimum guaranteed bandwidth for this queue.
<b>Step 5</b>	switch(config-pmap-que)# <b>exit</b>	Exits the current configuration mode.
<b>Step 6</b>	(Optional) switch(config)# <b>copy running-config startup-config</b>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

### Example

This example shows how to configuring packet shaping using 200000 packets per second (pps):

```

switch# configuration terminal
class-map type qos match-all cq
    match access-group name test
class-map type queuing cqu
    match qos-group 2
policy-map type qos pq
    class cq
        set qos-group 2
policy-map type queuing pqu
    class type queuing cqu
        random-detect minimum-threshold 100 packets maximum-threshold 350 packets
switch(config)# policy-map type queuing pqu
switch(config-pmap-que)# class type queuing cqu
switch(config-pmap-que)# shape pps 200000
switch(config-pmap-que)# exit
switch(config)# copy running-config startup-config

```

## Verifying Traffic Shaping

To display traffic shaping configuration information, perform one of the following tasks:

Command	Purpose
<b>show queuing</b>	Displays the queuing information configured for all interfaces. It includes shaper configuration information for each class, the control queue Tx and drop statistics for each port, and WRED drop packet counts.
<b>show queuing interface</b> <i>slot/port</i>	Displays the queuing information configured on the specified interface.
<b>show interface</b> <i>slot/port</i>	Shows the aggregated output traffic rate on all egress queues of the specified interface.

## Configuration Example for Traffic Shaping

The following example shows a sample configuration for traffic shaping using 200000 packets per second:

```

class-map type qos match-all cq
    match access-group name test
class-map type queuing cqu
    match qos-group 2
policy-map type qos pq
    class cq
        set qos-group 2
policy-map type queuing pqu
    class type queuing cqu
        random-detect minimum-threshold 100 packets maximum-threshold 350 packets
        shape pps 200000
        bandwidth percent 50
    class type queuing class-default
        bandwidth percent 50
class-map type network-qos cn

```

```
match qos-group 2
policy-map type network-qos pn
  class type network-qos cn
    congestion-control random-detect
  class type network-qos class-default
system qos
  service-policy type network-qos pn
  service-policy type queuing output pqu
  service-policy type qos input pq
```





## INDEX

### A

- attaching [40](#)
  - system service policy [40](#)

### B

- bandwidth [39](#)
  - multicast traffic [39](#)
  - unicast traffic [39](#)

### C

- changed information [1](#)
  - description [1](#)
- class maps [16](#)
  - configuring [16](#)
- classification policies [12](#)
  - ingress [12](#)
- Configuration examples for [70](#)
- configuring [25, 48](#)
  - interface service policies [48](#)
  - type network QoS policies [25](#)
- Configuring type queuing policies [26](#)
- configuring untagged CoS [47](#)
- CoS marking [34](#)
  - configuring [34](#)
- CPU traffic [14](#)
  - QoS [14](#)

### D

- default system service policies [41](#)
  - restoring [41](#)
- DSCP classification [19](#)
  - configuring [19](#)

### E

- ecn threshold, configuring [29](#)
  - qos [29](#)
- enabling [41](#)
  - jumbo MTU [41](#)
- Enabling PFC on a traffic class [64](#)

### I

- information about [6, 7](#)
  - policy types [7](#)
  - system classes [6](#)
- ingress [12](#)
  - classification policies [12](#)
- IP precedence marking [36](#)
  - qos [36](#)

### J

- jumbo MTU [42](#)
  - verifying [42](#)

### M

- marking [34](#)
  - information about [34](#)
- modular QoS CLI [5](#)
- MQC [5](#)
- multicast traffic [39](#)
  - changing bandwidth [39](#)

### N

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

### O

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

### P

- packet buffer, monitoring [58](#)
  - qos [58](#)
- PFC [62, 63, 64, 66, 69, 70](#)
  - Configuration examples for [70](#)
  - Configuring [66](#)
  - Configuring PFC [66](#)
  - default settings [63](#)
  - enabling [64](#)

PFC (*continued*)

- guidelines and limitations [62](#)
- Verifying the configuration [69](#)
- PFC frame counter statistics, monitoring [69](#)
- policy maps [22](#)
  - creating [22](#)
- policy types [7](#)
  - information about [7](#)
- precedence classification [21](#)
  - configuring [21](#)
- Priority flow control; PFC [61](#)
- priority groups [13](#)
- priority groups, configuring [32](#)
- priority-flow-control watch-dog interval [67](#)
- priority-flow-control watch-dog shutdown-multiplier [68](#)
- priority-flow-control watch-dog-interval [67](#)

**Q**

- qos [29, 36](#)
  - ecn threshold, configuring [29](#)
  - IP precedence marking [36](#)
- Qos [14](#)
  - CPU traffic [14](#)
- QoS [19, 21](#)
  - DSCP classification [19](#)
    - configuring [19](#)
  - precedence classification [21](#)
    - configuring [21](#)
- quality of service [5](#)
  - overview [5](#)
- Quality of Service, *See* QoS

**R**

- restoring [41](#)
  - default system service policies [41](#)

**S**

- shaping [71](#)
  - traffic [71](#)
- shaping traffic [72](#)
  - guidelines and limitations [72](#)
  - procedure [72](#)
- system classes [6](#)
  - information about [6](#)
- system service policy [40](#)
  - attaching [40](#)

**T**

- traffic shaping [71, 72](#)
  - configuring [72](#)
  - guidelines and limitations [72](#)
  - information about [71](#)
- type QoS polices, configuring [24](#)

**U**

- unicast traffic [39](#)
  - changing bandwidth [39](#)

**V**

- Verifying the PFC configuration [69](#)