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.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

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: 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. (1721R)

© 2018–2019 Cisco Systems, Inc. All rights reserved.
Configuring System Classes 14
Configuring Class Maps 14
  Configuring ACL Classification 16
  Configuring CoS Classification 17
  Configuring DSCP Classification 17
  Configuring IP RTP Classification 19
  Configuring RDMA over Converged Ethernet (RoCE) Classification 20
  Configuring Precedence Classification 21
Creating Policy Maps 22
  Configuring Type QoS Policies 23
  Configuring Type Network QoS Policies 24
  Configuring Type Queuing Policies 25
Information About Marking 26
  Configuring DSCP Marking 27
  Configuring IP Precedence Marking 28
Attaching the System Service Policy 30
Restoring the Default System Service Policies 31
Enabling the Jumbo MTU 31
Verifying the Jumbo MTU 32
Configuring QoS on Interfaces 34
  Configuring Untagged CoS 34
Configuring Buffers and Queues 34
  Configuring a Multicast Slow Receiver Port 34
  Configuring the Percentage of Buffer Used for a Specific QoS Group or Virtual Lane 35
  Configuring the Percentage of Buffer Used for SPAN Traffic 36
Verifying the QoS Configuration 37

CHAPTER 4

Configuring Active Latency Monitoring 45
  Active Latency Monitoring Overview 45
  Active Latency Monitoring Guidelines and Limitations 45
  Configuring Active Latency Monitoring 46
  Show Examples for Active Latency Monitoring 47

CHAPTER 5

Configuring Link Level Flow Control 49
Link Level Flow Control  49
Guidelines and Limitations for Link Level Flow Control  49
Information About Link Level Flow Control  50
   Link Level Flow Control on Interfaces  50
   Link Level Flow Control on Ports  50
   Mismatched Link Level Flow Control Configurations  50
How to Configure Link Level Flow Control  51
   Configuring Link Level Flow Control Receive  51
   Configuring Link Level Flow Control Transmit  52
Configuration Examples for Link Level Flow Control  54
   Example: Configuring Link Level Flow Control Receive  54

CHAPTER 6
Configuring Priority Flow Control  55
About Priority Flow Control  55
Licensing Requirements for Priority Flow Control  56
Prerequisites for Priority Flow Control  56
Guidelines and Limitations for Priority Flow Control  56
Default Settings for Priority Flow Control  58
Configuring Priority Flow Control  58
   Enabling Priority Flow Control on a Traffic Class  59
   Configuring Pause Buffer Thresholds  62
   Configuring Queue Limits  63
Verifying the Priority Flow Control Configuration  63
Configuration Examples for Priority Flow Control  64
Preface

The preface contains the following sections:

• Audience, on page vii
• Document Conventions, on page vii
• Documentation Feedback, on page viii

Audience

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

Document Conventions

As part of our constant endeavor to remodel our documents to meet our customers' requirements, we have modified the manner in which we document configuration tasks. As a result of this, you may find a deviation in the style used to describe these tasks, with the newly included sections of the document following the new format.

Command descriptions use the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold</strong></td>
<td>Bold text indicates the commands and keywords that you enter literally as shown.</td>
</tr>
<tr>
<td><em>Italic</em></td>
<td>Italic text indicates arguments for which the user supplies the values.</td>
</tr>
<tr>
<td>[x]</td>
<td>Square brackets enclose an optional element (keyword or argument).</td>
</tr>
<tr>
<td>[x</td>
<td>y]</td>
</tr>
<tr>
<td>{x</td>
<td>y}</td>
</tr>
</tbody>
</table>
Description

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[x {y</td>
<td>z}]</td>
</tr>
<tr>
<td>variable</td>
<td>Indicates a variable for which you supply values, in context where italics cannot be used.</td>
</tr>
<tr>
<td>string</td>
<td>A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.</td>
</tr>
</tbody>
</table>

Examples use the following conventions:

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

This document uses the following conventions:

**Note**

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

**Caution**

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

**Documentation Feedback**

To provide technical feedback on this document, or to report an error or omission, please send your comments to: .

We appreciate your feedback.
New and Changed Information in this Release

This chapter contains the following sections:

- New and Changed Information for this Release, on page 1

New and Changed Information for this Release

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

Table 1: New and Changed Quality of Service Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Added or Changed in Release</th>
<th>Where Documented</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title page</td>
<td>Changed the document title from 9.x to 9.2(x).</td>
<td></td>
<td>Title page</td>
</tr>
<tr>
<td>Priority Flow Control (PFC)</td>
<td>Added Priority Flow Control (PFC) support.</td>
<td>9.2(2)</td>
<td>Configuring Priority Flow Control, on page 55</td>
</tr>
<tr>
<td>No updates since Cisco NX-OS Release 7x</td>
<td>First 9x Release</td>
<td>9.2(1)</td>
<td></td>
</tr>
</tbody>
</table>
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.
- Network QoS policies include configuring maximum transmission unit (MTU).
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 14
• Configuring QoS on Interfaces, on page 34
• Configuring Buffers and Queues, on page 34
• Verifying the QoS Configuration, on page 37

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.

**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**

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

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

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

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

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

  - Congestion Control ECN—Data Center TCP (DCTCP) is an enhancement to the TCP congestion control algorithm for data center networks. It leverages Explicit Congestion Notification (ECN) feature, to mark all the packets when the queue length exceeds a configured ECN threshold value. The 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 dctcp ecn` command in the network-qos policy map mode.

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

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

The Cisco Nexus device supports type queuing in the egress direction.

**Note** Some configuration parameters when applied to an EtherChannel are not reflected on the configuration of the member ports.
On Cisco Nexus 3500 Series switches, QoS remarking will never work until a qos-group is defined under the qos policy. This is an expected behaviour that if there no qos-group is applied, it must get classified under the default queue.

- **Classification**—The traffic that matches this class are as follows:
  - **QoS Group**—A class map of type queuing identifies a system class and is matched by its associated QoS group.

- **Policy**—The actions that are performed on the matching traffic are as follows:

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

  - **Bandwidth**—Sets the guaranteed scheduling deficit weighted round robin (DWRR) percentage for the system class.
  - **Priority**—Sets a system class for strict-priority scheduling. Only one system class can be configured for priority in a given queuing policy.

- **Type qos**—A type QoS policy is used to classify traffic that is based on various Layer 2, Layer 3, and Layer 4 fields in the frame and to map it to system classes.

  - **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:

  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 3500 Series switches support: | • Five QoS groups |
| | • Five queues for unicast |
| | • Five queues for multicast |

**Network QoS Policy Type**

A network-qos policy is used to instantiate system classes and associate parameters with those classes that are of system-wide scope.

• Classification—The traffic that matches this class are as follows:
  • QoS Group—A class map of type network-qos identifies a system class and is matched by its associated qos-group.

• Policy—The actions that are performed on the matching traffic are as follows:

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

  Note
  All user defined classes have to be defined under network-qos policy and the network-qos policy has to be applied under "system qos".

• 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 all classes for all ports.

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

• Congestion Control DCTCP and ECN—Data Center TCP (DCTCP) is an enhancement to the TCP congestion control algorithm for data center networks. It leveragesExplicit Congestion Notification (ECN) feature, to mark all the packets when the queue length exceeds a configured DCTCP threshold value. The routers and end hosts use this marking as a signal that the network is congested to slow down sending packets.

To enable an DCTCP/ECN, use either the `congestion-control dctcp ecn-threshold threshold-bytes` command or the `congestion-control random-detect ecn` command in the network-qos policy-map mode.
Enabling DCTCP and ECN on a class on a network-qos policy implies that DCTCP and ECN is enabled for all ports in the system.

The following example shows how to enable DCTCP and ECN and verify the configuration on the network QoS policy maps:

```
switch# configuration terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# policy-map type network-qos system_network_policy
switch(config-pmap-nq)# class type network-qos nc1
switch(config-pmap-nq-c)# set cos 2
switch(config-pmap-nq-c)# class type network-qos nc2
switch(config-pmap-nq-c)# congestion-control dctcp ecn-threshold 30000 bytes
switch(config-pmap-nq-c)# class type network-qos nc3
switch(config-pmap-nq-c)# congestion-control random-detect ecn
switch(config-pmap-nq-c)# system qos
switch(config-sys-qos)# service-policy type network-qos system_network_policy
switch(config-sys-qos)# end
switch#
```

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

policy-map type network-qos system_network_policy
class type network-qos nc1
  match qos-group 1
  mtu 1500
  set cos 2
  class type network-qos nc2
  match qos-group 2
  mtu 1500
  congestion-control dctcp ecn-threshold 30000 bytes
  class type network-qos nc3
  match qos-group 3
  mtu 1500
  congestion-control random-detect ecn
  class type network-qos class-default
  match qos-group 0
  mtu 1500
```

When configuring LLFC/PFC, use the `pause no-drop/pfc-cos` command. For more information, see "Configuring Link Level Flow Control" and "Configuring Priority Flow Control".
Queuing Policy Type

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

The Cisco Nexus device supports type queuing in the egress direction.

Note

Some configuration parameters when applied to a Port Channel are not reflected on the configuration of the member ports.

Note

Queuing shaping functionality is not supported for the Nexus 3500.

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

Note

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

• Bandwidth—Sets the guaranteed scheduling deficit weighted round robin (DWRR) percentage for the system class.

• Priority—Sets a system class for strict-priority scheduling. Only one system class can be configured for priority in a given queuing policy.

QoS Policy Type

A QoS policy type 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 a Port Channel 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.

  Note
  Ingress/Egress policers are not supported in Nexus 3500.

• QoS Group—Sets the QoS group that corresponds to the system class this traffic flow is mapped to.
  • Cisco Nexus device supports the following:
    • Five QoS groups
    • Five queues for unicast
    • Five queues for multicast

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. No MTU configuration is required under non-default network-qos classes. MTU configuration CLI under non-default classes is blocked. MTU configuration under default class is implicitly applied to all user defined classes.

• 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. The device generates a syslog message to inform you of this requirement.
Trust Boundaries

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

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

- Any packet that is not tagged with an 802.1p CoS value is classified into the default drops 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.

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 five 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.

- 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. This queue is serviced before all other queues except the control traffic queue (which carries control rather than data traffic).

QoS for Traffic Directed to the CPU

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

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

• Switch resources (such as buffers, virtual output queues, and egress queues) are partitioned based on the default and user-defined system classes. Cisco NX-OS automatically adjusts the resource allocation to accommodate the configured system classes.

• When configuring Port Channels, the service policy configured on an Port Channel applies to all member interfaces.

• By default, queues 6 and 7 are reserved for control plane traffic and queue 5 for SPAN traffic. So you can configure four classes along with the default class.

• On Cisco Nexus 3548 Series switches, the bandwidth percentage that is configured under the queuing policy is not honored under the following conditions:
  • When there is an egress port that is experiencing congestion due to an input/output rate mismatch.
  • When there are multiple traffic classes using different UC/MC queues.
  • If all streams are competing for the buffers because the input rate of all streams is more than the output rate.

Some streams experience buffer crunch as the competing streams use up all the system buffers. There is no fair distribution of the shared buffers on Cisco Nexus N3548 Series switches, due to which the streams that can not be buffered are dropped aggressively. This results in the output rates that are less than the configured bandwidth for that stream and the other streams that exceed the configured bandwidth.

To work around this issue, you have to configure the CLI command `hardware profile buffer qos-group X threshold Y` where X is the qos-group number of the traffic that is exceeding the configured bandwidth and Y is the percentage of the shared buffers that can be used by the stream. The threshold value Y should be a small value, for example, 10 or 20. It can be fine tuned based on the burst absorption rate that is required at the same time honoring the bandwidth. The default threshold is 95%.

• For Weighted Round-Robin (wrr), the `wrr unicast-bandwidth bandwidth_in_percent` command is used to specify the total bandwidth allocated for unicast traffic. The default is 50%.

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.
### SUMMARY STEPS

1. `switch# configure terminal`
2. `switch(config)# class-map [type {network-qos | qos | queuing}] class-map name`
3. (Optional) `switch(config)# class-map [type qos] [match-all | match-any] class-map name`
4. (Optional) `switch(config)# no class-map [type {network-qos | qos | queuing}] class-name`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><code>switch# configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>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.</td>
</tr>
<tr>
<td>`switch(config)# class-map [type {network-qos</td>
<td>qos</td>
</tr>
</tbody>
</table>
| | - **network-qos**—Network-wide (global) mode. CLI prompt: `switch(config-cmap-nq)#`
| | - **qos**—Classification mode; this is the default mode. CLI prompt: `switch(config-cmap-qos)#`
| | - **queuing**—Queuing mode. CLI prompt: `switch(config-cmap-que)#`
| **Step 3** | Specifies that packets must match any or all criteria that is defined for a class map. |
| (Optional) `switch(config)# class-map [type qos] [match-all | match-any] class-map name` | |
| | - **match-all**—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). |
| | - **match-any**—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). |
| **Step 4** | Deletes the specified class map. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters. |
| (Optional) `switch(config)# no class-map [type {network-qos | qos | queuing}] class-name` | |
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.

**SUMMARY STEPS**

1. switch# configure terminal
2. switch(config)# class-map type qos class-name
3. switch(config-cmap-qos)# match access-group name acl-name
4. (Optional) switch(config-cmap-qos)# no match access-group name acl-name

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>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.</td>
</tr>
<tr>
<td>switch(config)# class-map type qos class-name</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Configures a traffic class by matching packets based on the acl-name. The permit and deny ACL keywords are ignored in the matching. Note You can only define a single ACL in a class map. You cannot add any other match criteria to a class with a match access-group defined.</td>
</tr>
<tr>
<td>switch(config-cmap-qos)# match access-group name acl-name</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Removes the match from the traffic class.</td>
</tr>
<tr>
<td>(Optional) switch(config-cmap-qos)# no match access-group name acl-name</td>
<td></td>
</tr>
</tbody>
</table>

**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.

SUMMARY STEPS

1. switch# configure terminal
2. switch(config)# class-map type qos class-name
3. switch(config-cmap-qos)# match cos cos-value
4. (Optional) switch(config-cmap-qos)# no match cos cos-value

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 switch# configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Step 2 switch(config)# class-map type qos class-name</td>
<td>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.</td>
</tr>
<tr>
<td>Step 3 switch(config-cmap-qos)# match cos cos-value</td>
<td>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.</td>
</tr>
<tr>
<td>Step 4 (Optional) switch(config-cmap-qos)# no match cos cos-value</td>
<td>Removes the match from the traffic class.</td>
</tr>
</tbody>
</table>

Example

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

```bash
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:

```bash
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.

Table 2: Standard DSCP Values

<table>
<thead>
<tr>
<th>Value</th>
<th>List of DSCP Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>af11</td>
<td>AF11 dscp (001010)—decimal value 10</td>
</tr>
<tr>
<td>Value</td>
<td>List of DSCP Values</td>
</tr>
<tr>
<td>-------</td>
<td>---------------------</td>
</tr>
<tr>
<td>af12</td>
<td>AF12 dscp (001100)—decimal value 12</td>
</tr>
<tr>
<td>af13</td>
<td>AF13 dscp (001110)—decimal value 14</td>
</tr>
<tr>
<td>af21</td>
<td>AF21 dscp (010010)—decimal value 18</td>
</tr>
<tr>
<td>af22</td>
<td>AF22 dscp (010100)—decimal value 20</td>
</tr>
<tr>
<td>af23</td>
<td>AF23 dscp (010110)—decimal value 22</td>
</tr>
<tr>
<td>af31</td>
<td>AF31 dscp (011010)—decimal value 26</td>
</tr>
<tr>
<td>af32</td>
<td>AF32 dscp (011100)—decimal value 28</td>
</tr>
<tr>
<td>af33</td>
<td>AF33 dscp (011110)—decimal value 30</td>
</tr>
<tr>
<td>af41</td>
<td>AF41 dscp (100010)—decimal value 34</td>
</tr>
<tr>
<td>af42</td>
<td>AF42 dscp (100100)—decimal value 36</td>
</tr>
<tr>
<td>af43</td>
<td>AF43 dscp (100110)—decimal value 38</td>
</tr>
<tr>
<td>cs1</td>
<td>CS1 (precedence 1) dscp (001000)—decimal value 8</td>
</tr>
<tr>
<td>cs2</td>
<td>CS2 (precedence 2) dscp (010000)—decimal value 16</td>
</tr>
<tr>
<td>cs3</td>
<td>CS3 (precedence 3) dscp (011000)—decimal value 24</td>
</tr>
<tr>
<td>cs4</td>
<td>CS4 (precedence 4) dscp (100000)—decimal value 32</td>
</tr>
<tr>
<td>cs5</td>
<td>CS5 (precedence 5) dscp (101000)—decimal value 40</td>
</tr>
<tr>
<td>cs6</td>
<td>CS6 (precedence 6) dscp (110000)—decimal value 48</td>
</tr>
<tr>
<td>cs7</td>
<td>CS7 (precedence 7) dscp (111000)—decimal value 56</td>
</tr>
<tr>
<td>default</td>
<td>Default dscp (000000)—decimal value 0</td>
</tr>
<tr>
<td>ef</td>
<td>EF dscp (101110)—decimal value 46</td>
</tr>
</tbody>
</table>

### SUMMARY STEPS

1. switch# configure terminal
2. switch(config)# class-map type qos class-name
3. switch(config-cmap-qos)# match dscp dscp-list
4. (Optional) switch(config-cmap-qos)# no match dscp dscp-list

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 switch# configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>
### Purpose

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td>switch(config)# class-map type qos class-name</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>switch(config-cmap-qos)# match dscp dscp-list</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>(Optional) switch(config-cmap-qos)# no match dscp dscp-list</td>
</tr>
</tbody>
</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.

### SUMMARY STEPS

1. switch# configure terminal
2. switch(config)# class-map type qos class-name
3. switch(config-cmap-qos)# match ip rtp port-number
4. (Optional) switch(config-cmap-qos)# no match ip rtp port-number

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>switch# configure terminal</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>switch(config)# class-map type qos class-name</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>switch(config-cmap-qos)# match ip rtp port-number</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>(Optional) switch(config-cmap-qos)# no match ip rtp port-number</td>
</tr>
<tr>
<td></td>
<td>Removes the match from the traffic class.</td>
</tr>
</tbody>
</table>

**Example**

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

```plaintext
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:

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

### Configuring RDMA over Converged Ethernet (RoCE) Classification

The following is how to configure for the ROCE protocol,

#### Note

When configuring RoCE, the port list ranges from 2000 to 65535. The recommended port to use is 3804.

### SUMMARY STEPS

1. switch# configure terminal
2. switch(config)# class-map type qos class-name
3. switch(config-cmap-qos)# match ip roce port-number
4. (Optional) switch(config-cmap-qos)# no match ip roce port-number

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>switch# configure terminal</td>
</tr>
<tr>
<td></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>switch(config)# class-map type qos class-name</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>switch(config-cmap-qos)# match ip roce port-number</td>
</tr>
<tr>
<td></td>
<td>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. The recommended port is 3804.</td>
</tr>
</tbody>
</table>
Configuring Precedence Classification

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

Table 3: Precedence Values

<table>
<thead>
<tr>
<th>Value</th>
<th>List of Precedence Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;0-7&gt;</td>
<td>IP precedence value</td>
</tr>
<tr>
<td>critical</td>
<td>Critical precedence (5)</td>
</tr>
<tr>
<td>flash</td>
<td>Flash precedence (3)</td>
</tr>
<tr>
<td>flash-override</td>
<td>Flash override precedence (4)</td>
</tr>
<tr>
<td>immediate</td>
<td>Immediate precedence (2)</td>
</tr>
<tr>
<td>internet</td>
<td>Internetwork control precedence (6)</td>
</tr>
<tr>
<td>network</td>
<td>Network control precedence (7)</td>
</tr>
<tr>
<td>priority</td>
<td>Priority precedence (1)</td>
</tr>
<tr>
<td>routine</td>
<td>Routine precedence (0)</td>
</tr>
</tbody>
</table>

**SUMMARY STEPS**

1. switch# configure terminal
2. switch(config)# class-map type qos match-any class-name
3. switch(config-cmap-qos)#match precedence precedence-values
4. (Optional) switch((config-cmap-qos)# no match precedence precedence-values

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch# configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>switch(config)# class-map type qos match-any class-name</td>
<td>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.</td>
</tr>
<tr>
<td>switch(config-cmap-qos)#match precedence precedence-values</td>
<td>Configures the traffic class by matching packets based on precedence values. For a list of precedence values, see the Precedence Values table.</td>
</tr>
</tbody>
</table>
### Purpose

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 4</strong></td>
<td>(Optional) switch((config-cmap-qos)# no match precedence precedence-values)</td>
</tr>
</tbody>
</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:

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

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

```plaintext
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

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.

### SUMMARY STEPS

1. switch# configure terminal
2. switch(config)# policy-map [type {network-qos | qos | queuing}] policy-name
3. (Optional) switch(config)# no policy-map [type {network-qos | qos | queuing}] policy-name
4. switch(config-pmap)# class [type {network-qos | qos | queuing}] class-name
5. (Optional) switch(config-pmap)# no class [type {network-qos | qos | queuing}] class-name
DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>switch# configure terminal</td>
</tr>
</tbody>
</table>
| Step 2 | switch(config)# policy-map [type {network-qos | qos | queuing}] policy-name | 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:  
  • network-qos—Network-wide (global) mode. CLI prompt: switch(config-pmap-nq)#  
  • qos—Classification mode; this is the default mode. CLI prompt: switch(config-pmap-qos)#  
  • queuing—Queuing mode. CLI prompt: switch(config-pmap-que)# |
| Step 3 | (Optional) switch(config)# no policy-map [type {network-qos | qos | queuing}] policy-name | Deletes the specified policy map. |
| Step 4 | switch(config-pmap)# class [type {network-qos | qos | queuing}] class-name | Associates a class map with the policy map, and enters configuration mode for the specified system class. The three class-map configuration modes are as follows:  
  • network-qos—Network-wide (global) mode. CLI prompt: switch(config-pmap-c-nq)#  
  • qos—Classification mode; this is the default mode. CLI prompt: switch(config-pmap-c-qos)#  
  • queuing—Queuing mode. CLI prompt: switch(config-pmap-c-que)#  
  **Note** The associated class map must be the same type as the policy-map type. |
| Step 5 | (Optional) switch(config-pmap)# no class [type {network-qos | qos | queuing}] class-name | Deletes the class map association. |

Configuring Type QoS Policies

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

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

SUMMARY STEPS

1. switch# configure terminal
2. switch(config)# policy-map type qos policy-name
### 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.

#### SUMMARY STEPS

1. switch# configure terminal
2. switch(config)# policy-map type network-qos policy-name
3. switch(config-pmap-nq)# class type network-qos class-name
4. switch(config-pmap-c-nq)# mtu mtu-value
5. (Optional) switch(config-pmap-c-nq)# no mtu
6. switch(config-pmap-c-nq)# set cos cos-value
7. (Optional) switch(config-pmap-c-nq)# no set cos cos-value

---

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>switch# configure terminal</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>switch(config)# policy-map type qos policy-name</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>switch(config-pmap-qos)# [class</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>switch(config-pmap-c-qos)# set qos-group qos-group-value</td>
</tr>
</tbody>
</table>

#### 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
```
**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>switch# configure terminal</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>switch(config)# policy-map type network-qos policy-name</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>switch(config-pmap-nq)# class type network-qos class-name</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>switch(config-pmap-c-nq)# mtu mtu-value</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>(Optional) switch(config-pmap-c-nq)# no mtu</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>switch(config-pmap-c-nq)# set cos cos-value</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>(Optional) switch(config-pmap-c-nq)# no set cos cos-value</td>
</tr>
</tbody>
</table>

**Example**

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

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

**Configuring Type Queuing Policies**

Type queuing policies are used for scheduling and buffering the traffic of a specific system class. A type queuing policy is identified by its QoS group and can be attached to the system or to individual interfaces (except for Fabric Extender host interfaces) for input or output traffic.

**SUMMARY STEPS**

1. switch# configure terminal
2. switch(config)# policy-map type queuing policy-name
3. switch(config-pmap-que)# class type queuing class-name
4. switch(config-pmap-c-que)# priority
5. (Optional) switch(config-pmap-c-que)# no priority
6. `switch(config-pmap-c-que)# bandwidth percent percentage`
7. (Optional) `switch(config-pmap-c-que)# no bandwidth percent percentage`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>switch# configure terminal</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>switch(config)# policy-map type queuing policy-name</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>switch(config-pmap-que)# class type queuing class-name</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>switch(config-pmap-c-que)# priority</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>(Optional) switch(config-pmap-c-que)# no priority</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>switch(config-pmap-c-que)# bandwidth percent percentage</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>(Optional) switch(config-pmap-c-que)# no bandwidth percent percentage</td>
</tr>
</tbody>
</table>

**Example**

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

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

**Information About Marking**

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

## Configuring DSCP Marking

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

---

**Note**

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

---

### Table 4: Standard DSCP Values

<table>
<thead>
<tr>
<th>Value</th>
<th>List of DSCP Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>af11</td>
<td>AF11 dscp (001010)—decimal value 10</td>
</tr>
<tr>
<td>af12</td>
<td>AF12 dscp (001100)—decimal value 12</td>
</tr>
<tr>
<td>af13</td>
<td>AF13 dscp (001110)—decimal value 14</td>
</tr>
<tr>
<td>af21</td>
<td>AF21 dscp (010010)—decimal value 18</td>
</tr>
<tr>
<td>af22</td>
<td>AF22 dscp (010100)—decimal value 20</td>
</tr>
<tr>
<td>af23</td>
<td>AF23 dscp (010110)—decimal value 22</td>
</tr>
<tr>
<td>af31</td>
<td>AF31 dscp (011010)—decimal value 26</td>
</tr>
<tr>
<td>af32</td>
<td>AF40 dscp (011100)—decimal value 28</td>
</tr>
<tr>
<td>af33</td>
<td>AF33 dscp (011110)—decimal value 30</td>
</tr>
<tr>
<td>af41</td>
<td>AF41 dscp (100010)—decimal value 34</td>
</tr>
<tr>
<td>af42</td>
<td>AF42 dscp (100100)—decimal value 36</td>
</tr>
<tr>
<td>af43</td>
<td>AF43 dscp (100110)—decimal value 38</td>
</tr>
<tr>
<td>cs1</td>
<td>CS1 (precedence 1) dscp (001000)—decimal value 8</td>
</tr>
<tr>
<td>cs2</td>
<td>CS2 (precedence 2) dscp (010000)—decimal value 16</td>
</tr>
<tr>
<td>cs3</td>
<td>CS3 (precedence 3) dscp (011000)—decimal value 24</td>
</tr>
<tr>
<td>cs4</td>
<td>CS4 (precedence 4) dscp (100000)—decimal value 32</td>
</tr>
<tr>
<td>Value</td>
<td>List of DSCP Values</td>
</tr>
<tr>
<td>-------</td>
<td>---------------------</td>
</tr>
<tr>
<td>cs5</td>
<td>CS5 (precedence 5)</td>
</tr>
<tr>
<td></td>
<td>dscp (101000)</td>
</tr>
<tr>
<td></td>
<td>— decimal value 40</td>
</tr>
<tr>
<td>cs6</td>
<td>CS6 (precedence 6)</td>
</tr>
<tr>
<td></td>
<td>dscp (110000)</td>
</tr>
<tr>
<td></td>
<td>— decimal value 48</td>
</tr>
<tr>
<td>cs7</td>
<td>CS7 (precedence 7)</td>
</tr>
<tr>
<td></td>
<td>dscp (111000)</td>
</tr>
<tr>
<td></td>
<td>— decimal value 56</td>
</tr>
<tr>
<td>default</td>
<td>Default dscp (000000)</td>
</tr>
<tr>
<td></td>
<td>— decimal value 0</td>
</tr>
<tr>
<td>ef</td>
<td>EF dscp (101110)</td>
</tr>
<tr>
<td></td>
<td>— decimal value 46</td>
</tr>
</tbody>
</table>

**SUMMARY STEPS**

1. configure terminal
2. policy-map type qos qos-policy-map-name
3. class [type qos] {class-map-name | class-default}
4. set dscp dscp-value

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Step 2 policy-map type qos qos-policy-map-name</td>
<td>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.</td>
</tr>
<tr>
<td>Step 3 class [type qos] {class-map-name</td>
<td>class-default}</td>
</tr>
<tr>
<td>Step 4 set dscp dscp-value</td>
<td>Sets the DSCP value to dscp-value. See the Standards DSCP Values table.</td>
</tr>
</tbody>
</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. The following table shows the precedence values:
You can set IP precedence or DSCP but you cannot set both values because they modify the same field in the IP packet.

### Table 5: Precedence Values

<table>
<thead>
<tr>
<th>Value</th>
<th>List of Precedence Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-7</td>
<td>IP precedence value</td>
</tr>
<tr>
<td>critical</td>
<td>Critical precedence (5)</td>
</tr>
<tr>
<td>flash</td>
<td>Flash precedence (3)</td>
</tr>
<tr>
<td>flash-override</td>
<td>Flash override precedence (4)</td>
</tr>
<tr>
<td>immediate</td>
<td>Immediate precedence (2)</td>
</tr>
<tr>
<td>internet</td>
<td>Internetwork control precedence (6)</td>
</tr>
<tr>
<td>network</td>
<td>Network control precedence (7)</td>
</tr>
<tr>
<td>priority</td>
<td>Priority precedence (1)</td>
</tr>
<tr>
<td>routine</td>
<td>Routine precedence (0)</td>
</tr>
</tbody>
</table>

### SUMMARY STEPS

1. `config terminal`
2. `policy-map [type qos] qos-policy-map-name`
3. `class [type qos] {class-map-name | class-default}`
4. `set precedence precedence-value`

### DETAILED STEPS

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

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

#### SUMMARY STEPS

1. `switch# configure terminal`
2. `switch(config)# system qos`
3. `switch(config-sys-qos)# service-policy type {network-qos | qos input [queuing [input | output]]} policy-name`

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>switch# configure terminal</code></td>
<td>Enters global configuration mode.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>switch(config)# system qos</code></td>
<td>Enters system class configuration mode.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> `switch(config-sys-qos)# service-policy type {network-qos</td>
<td>qos input [queuing [input</td>
<td>output]]} policy-name`</td>
</tr>
<tr>
<td></td>
<td>• network-qos—Network-wide (system qos) mode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• qos—Classification mode (system qos input or interface input only).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• queuing—Queuing mode (output at system qos and interface).</td>
<td></td>
</tr>
</tbody>
</table>

**Note** There is no default policy-map configuration mode; you must specify the **type**. The **input** keyword specifies that this policy map should be applied to traffic received on an interface. The **output** keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply **input** to a qos policy; you can only apply **output** to a queuing policy.

---

**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)#
```
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.

**SUMMARY STEPS**

1. switch# configure terminal
2. switch(config)# system qos
3. switch(config-sys-qos)# no service-policy type qos input policy-map name
4. switch(config-sys-qos)# no service-policy type network-qos policy-map name
5. switch(config-sys-qos)# no service-policy type queuing output policy-map name

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>switch# configure terminal</td>
</tr>
<tr>
<td>Step 2</td>
<td>switch(config)# system qos</td>
</tr>
<tr>
<td>Step 3</td>
<td>switch(config-sys-qos)# no service-policy type qos input policy-map name</td>
</tr>
<tr>
<td>Step 4</td>
<td>switch(config-sys-qos)# no service-policy type network-qos policy-map name</td>
</tr>
<tr>
<td>Step 5</td>
<td>switch(config-sys-qos)# no service-policy type queuing output policy-map name</td>
</tr>
</tbody>
</table>

**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
```

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

This example shows how to change qos to enable the jumbo MTU:
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. To verify the jumbo MTU for Ethernet traffic, use the `show queueing interface ethernet slot/chassis_number` command and find "HW MTU" in the command output to check the MTU for QoS group 0. The value should be 9216.

The `show interface` command always displays 1500 as the MTU. Because the Cisco Nexus device supports different MTUs for different QoS groups, it is not possible to represent the MTU as one value on a per interface level.

This example shows how to display jumbo MTU information:

```
switch# sh queuing interface ethernet 1/1

 HW MTU of Ethernet1/1 : 1500 bytes

```

```
Egress Queuing for Ethernet1/1 [System]

<table>
<thead>
<tr>
<th>QoS-Group#</th>
<th>Bandwidth%</th>
<th>PrioLevel</th>
<th>Shape MIn</th>
<th>Shape Max</th>
<th>Units</th>
<th>QLimit</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4969339(S)</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4969339(S)</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4969339(S)</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4969339(S)</td>
</tr>
<tr>
<td>0</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4969339(S)</td>
</tr>
</tbody>
</table>

Mcast pkts dropped : 0
```

```
+-------------------------------------------------------------------+
| QOS GROUP 0 | | | | | |
| +-----------------------------+-----------------------------+-----------------------------+|
| | Unicast | OOBFC Unicast | Multicast | |
| | Dropped Pkts | 0 | 0 | 0 | |
| +-----------------------------+-----------------------------+-----------------------------+|
| QOS GROUP 1 | |
| +-----------------------------+-----------------------------+-----------------------------+|
| | Unicast | OOBFC Unicast | Multicast | |
| | Dropped Pkts | 0 | 0 | 0 | |
| +-----------------------------+-----------------------------+-----------------------------+|
```

```
```

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.
Ingress Queuing for Ethernet1/1

<table>
<thead>
<tr>
<th>QoS-Group#</th>
<th>Buff Size</th>
<th>Pause Th</th>
<th>Resume Th</th>
<th>QLimit</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
</tbody>
</table>

PFC Statistics

TxPPP: 0, RxPPP: 0

<table>
<thead>
<tr>
<th>COS QoS Group</th>
<th>TxCount</th>
<th>RxCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>0</td>
</tr>
</tbody>
</table>
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.

**SUMMARY STEPS**

1. switch# configure terminal
2. switch(config)# interface {ethernet [chassis/]slot/port | port-channel channel-number}
3. switch(config-if)# untagged cos cos-value

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>switch# configure terminal</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>switch(config)# interface {ethernet [chassis/]slot/port</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>switch(config-if)# untagged cos cos-value</td>
</tr>
</tbody>
</table>

**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 Buffers and Queues

Configuring a Multicast Slow Receiver Port

When you have a combination of 10-gigabyte and 1-gigabyte ports, you can use this command on the 1-gigabyte port to reduce the effects of the 1-gigabyte port blocking the 10-gigabyte port. Use this command on the 1-gigabyte port only when there is Head-of-Line Blocking (HOLB) on the 10-gigabyte port due to a slow receiver on the 1-gigabyte port.
Configuring QoS

Configuring the Percentage of Buffer Used for a Specific QoS Group or Virtual Lane

You can configure the percentage of shared buffer used for a specific QoS group or virtual lane (VL)

**SUMMARY STEPS**

1. switch# configure terminal
2. switch(config)# hardware profile buffer qosgroup number threshold percentage
3. (Optional) switch(config)# copy running-config startup-config

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> switch# configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong> switch(config)# hardware profile buffer qosgroup number threshold percentage</td>
<td>Configures the buffer for the specified QoS group. The number argument specifies the QoS group number. The range is from 0 to 4. The percentage argument specified</td>
</tr>
<tr>
<td><strong>Step 3</strong> (Optional) switch(config)# copy running-config startup-config</td>
<td>Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.</td>
</tr>
</tbody>
</table>
### Configuring the Percentage of Buffer Used for SPAN Traffic

You can configure the percentage of shared buffer used for SPAN traffic.

**SUMMARY STEPS**

1. `switch# configure terminal`
2. `switch# hardware profile buffer span-threshold percentage`
3. (Optional) `switch(config)# copy running-config startup-config`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><code>switch# configure terminal</code></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>switch# hardware profile buffer span-threshold percentage</code></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>(Optional) <code>switch(config)# copy running-config startup-config</code></td>
</tr>
</tbody>
</table>

**Example**

The following example shows how to configure 30 percent of the hardware buffer for SPAN traffic:

```
switch# configure terminal
switch(config)# hardware profile buffer span-threshold 30
switch(config)# copy running-config startup-config
```
## Verifying the Qos Configuration

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

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>switch# show class-map</code></td>
<td>Displays the class maps defined on the device.</td>
</tr>
<tr>
<td><code>switch# show policy-map [name]</code></td>
<td>Displays the policy maps defined on the device. Optionally, you can display the named policy only.</td>
</tr>
<tr>
<td><code>switch# show policy-map interface [interface number]</code></td>
<td>Displays the policy map settings for an interface or all interfaces.</td>
</tr>
<tr>
<td><code>switch# show policy-map system</code></td>
<td>Displays the policy map settings attached to the system qos.</td>
</tr>
<tr>
<td>`switch# show policy-map type {network-qos</td>
<td>qos</td>
</tr>
<tr>
<td><code>switch# show interface [interface slot/port] priority-flow-control [module number] [detail]</code></td>
<td>Displays the priority flowcontrol details for a specified interface.</td>
</tr>
<tr>
<td><code>switch# show interface untagged-cos [module number]</code></td>
<td>Displays the untagged CoS values for all interfaces.</td>
</tr>
<tr>
<td><code>switch# show running-config ipqos</code></td>
<td>Displays information about the running configuration for QoS.</td>
</tr>
<tr>
<td><code>switch# show startup-config ipqos</code></td>
<td>Displays information about the startup configuration for QoS.</td>
</tr>
<tr>
<td><code>switch# show queuing interface ethernet slot-no/port-no</code></td>
<td>Displays the queuing information on interfaces.</td>
</tr>
</tbody>
</table>

This example shows how to configure a network QoS policy:

```
switch(config)# class-map type network-qos cnq1
switch(config-cmap-ng)# match qos-group 1
switch(config-cmap-ng)# exit
switch(config)# class-map type network-qos cnq2
switch(config-cmap-ng)# match qos-group 2
switch(config-cmap-ng)# exit
switch(config)# policy-map type network-qos pnqos
switch(config-pmap-ng)# class type network-qos cnq1
switch(config-pmap-ng-c)# set cos 4
switch(config-pmap-ng-c)# exit
switch(config-pmap-ng)# class type network-qos cnq2
switch(config-pmap-ng-c)# set cos 5
switch(config-pmap-ng-c)# congestion-control random-detect ecn
switch(config-pmap-ng-c)# exit
switch(config-pmap-ng)# class type network-qos class-default
switch(config-pmap-ng-c)# mtu 9216
switch(config-pmap-ng-c)# exit
switch(config-pmap-ng)# exit
```
switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos pnqos
switch(config-sys-qos)#

This example shows how to configure a queuing policy:

switch(config)# class-map type queuing cqu1
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 2
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-que)# exit
switch(config)# policy-map type queuing pqu
switch(config-pmap-que)# class type queuing cqu1
switch(config-pmap-c-que)# bandwidth percent 10
switch(config-pmap-que)# exit
switch(config)# policy-map type queuing pqu
switch(config-pmap-que)# class type queuing cqu6
switch(config-pmap-c-que)# bandwidth percent 20
switch(config-pmap-que)# exit
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing output pqu
switch(config-sys-qos)#

This example shows how to configure a QoS policy:

switch(config)# class-map type qos cqos1
switch(config-cmap-qos)# match cos 1
switch(config-cmap-qos)# exit
switch(config)# class-map type qos cqos6
switch(config-cmap-qos)# match cos 6
switch(config-cmap-qos)# exit
switch(config)# policy-map type qos pqos
switch(config-pmap-qos)# class type qos cqos1
switch(config-pmap-c-qos)# set qos-group 1
switch(config-pmap-qos)# exit
switch(config)# policy-map type qos pqos
switch(config-pmap-qos)# class type qos cqos6
switch(config-pmap-c-qos)# set qos-group 2
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# show interface untagged-cos
Legend: * - On conversion to L2 interface
-------------------------------------
Interface Untagged-CoS
-------------------------------------
  Eth1/10  3
  Eth1/11  4
switch#

This example shows how to display the QoS running configuration:

switch(config)# show running-config ipqos

!Command: show running-config ipqos!Running configuration last done at: Tue Oct 16 06:59:37 2018
!Time: Tue Oct 16 07:00:15 2018

version 9.2(2) Bios:version 5.1.0
class-map type qos match-all cqos1
  match cos 1
class-map type qos match-all cqos6
  match cos 6
class-map type queuing cqu1
  match qos-group 1
class-map type queuing cqu6
  match qos-group 2
policy-map type qos pqos
  class cqos1
    set qos-group 1
class cqos6
    set qos-group 2
policy-map type queuing pqu
  class type queuing cqu1
    bandwidth percent 10
  class type queuing cqu6
    bandwidth percent 20
  class type queuing class-default
    bandwidth percent 70
class-map type network-qos cnq1
  match qos-group 1
class-map type network-qos cnq2
  match qos-group 2
policy-map type network-qos pnqos
  class type network-qos cnq1
    set cos 4
class type network-qos cnq2
    set cos 5
    congestion-control random-detect ecn
class type network-qos class-default
  mtu 9216
system qos
  service-policy type qos input pqos
  service-policy type network-qos pnqos
  service-policy type queuing output pqu

interface Ethernet1/1
  untagged cos 4

interface Ethernet1/3
  untagged cos 5

switch(config)#

This example shows how to display the 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 cqos2
  match cos 2

class-map type qos match-any class-default
  match any
Type queuing class-maps
----------------------------------------
class-map type queuing cqu1
  match qos-group 1

class-map type queuing cqu2
  match qos-group 2

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 cnq2
  match qos-group 2

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 cqos2
    set qos-group 2
  class type qos class-default
    set qos-group 0

policy-map type qos default-in-policy
  class type qos class-default
    set qos-group 0

Type queuing policy-maps
----------------------------------------
policy-map type queuing pqu
  class type queuing cqu1
    bandwidth percent 10
  class type queuing cqu2
    bandwidth percent 20
  class type queuing class-default
    bandwidth percent 70

policy-map type queuing default-out-policy
  class type queuing class-default
    bandwidth percent 100

Type network-qos policy-maps
```
policy-map type network-qos pnqos
  class type network-qos cnq1
    mtu 1500
    set cos 4
  class type network-qos cnq2
    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 1500
    set cos 4
  class type network-qos cnq2   match qos-group 2
    mtu 1500
    set cos 5
    congestion-control random-detect ecn
  class type network-qos class-default   match qos-group 0
    mtu 9216

Service-policy (qos) input: pqos
  policy statistics status: disabled
  Class-map (qos): cqos1 (match-all)
    Match: cos 1
    set qos-group 1
  Class-map (qos): cqos2 (match-all)
    Match: cos 2
    set qos-group 2
  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): cqu1 (match-any)
    Match: qos-group 1
    bandwidth percent 10
  Class-map (queuing): cqu6 (match-any)
    Match: qos-group 2
    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 : disabled

Ethernet1/1

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

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

Class-map (qos): cqos2 (match-all)
Match: cos 2
set qos-group 2

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): cqu1 (match-any)
Match: qos-group 1
bandwidth percent 10

Class-map (queuing): cqu2 (match-any)
Match: qos-group 2
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 for a specific interface:

switch# sh queuing interface ethernet 1/1

slot 1
-------

HW MTU of Ethernet1/1 : 1500 bytes

Egress Queuing for Ethernet1/1 [System]

<table>
<thead>
<tr>
<th>QoS-Group</th>
<th>Bandwidth</th>
<th>PrioLevel</th>
<th>Shape Min</th>
<th>Shape Max</th>
<th>Units</th>
<th>QLimit</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4969339(S)</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4969339(S)</td>
</tr>
</tbody>
</table>
Mcast pkts dropped : 0

<table>
<thead>
<tr>
<th>QOS GROUP 0</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS GROUP 1</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS GROUP 2</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS GROUP 3</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS GROUP 4</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS GROUP 5</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS GROUP 6</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS GROUP 7</th>
<th>Unicast</th>
<th>OOBFC Unicast</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped Pkts</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Ingress Queuing for Ethernet1/1

<table>
<thead>
<tr>
<th>QoS-Group#</th>
<th>Buff Size</th>
<th>Pause</th>
<th>Pause Th</th>
<th>Resume Th</th>
<th>QLimit</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0(S)</td>
</tr>
</tbody>
</table>
Verifying the Qos Configuration

<table>
<thead>
<tr>
<th>COS</th>
<th>QOS Group</th>
<th>TxCount</th>
<th>RxCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>-</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

PFC Statistics

```
switch#
```
Configuring Active Latency Monitoring

This chapter contains the following sections:

- Active Latency Monitoring Overview, on page 45
- Active Latency Monitoring Guidelines and Limitations, on page 45
- Configuring Active Latency Monitoring, on page 46
- Show Examples for Active Latency Monitoring, on page 47

Active Latency Monitoring Overview

Active Latency Monitoring provides a real-time view of the latency that is incurred by the packets while traveling through the switch on a per port basis. The latency measurement is FIFO measurement. Functionally, as soon as the packet enters the switch, the ASIC adds a timestamp to it. When it is scheduled to go out of the egress port, the egress port calculates the latency for each packet that is going out of that port based on current time and the ingress timestamp on the packet.

Note

Active latency monitoring is currently not available for Cisco Nexus N3548 Series switches. This feature is only supported for Cisco Nexus N3548-X Series switches.

Each egress port maintains the information in the frame count and the latency register, along with the minimum and maximum latency on that port. The software periodically reads the frame count (default 3 seconds) and the total latency to calculate the average latency per port. Based on per port latency information, the software calculates the average switch latency.

Active Latency Monitoring Guidelines and Limitations

Active Latency Monitoring has the following limitations and guidelines:

- Disabling the latency monitor does not clear the existing latency monitor data.
- Clear the latency monitor data before enabling the latency monitor.
- The latency monitor data is lost when the sampling interval is modified.
- The latency monitor data is not maintained across a switch reload.
Configuring Active Latency Monitoring

To configure active latency monitoring, complete the following steps:

**Note**
The average or maximum latency threshold is in nanoseconds. The software sampling interval value is between 1 to 30 seconds. The default values for the parameters are:

- Sampling = 3 seconds
- Threshold-avg = 1000000 nanoseconds
- Threshold-max = 2000000 nanoseconds

**SUMMARY STEPS**

1. clear hardware profile latency monitor
2. [no] hardware profile latency monitor
3. (Optional) hardware profile latency monitor threshold-avg <value>
4. (Optional) hardware profile latency monitor threshold-max <value>
5. (Optional) hardware profile latency monitor sampling <value>
6. exit
7. (Optional) show hardware profile latency monitor summary

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 clear hardware profile latency monitor</td>
<td>Clears the latency monitor data.</td>
</tr>
<tr>
<td>Step 2 [no] hardware profile latency monitor</td>
<td>Enables or disables latency monitoring.</td>
</tr>
<tr>
<td>Step 3 (Optional) hardware profile latency monitor threshold-avg &lt;value&gt;</td>
<td>Sets the average threshold for syslog generation. Range is 300 to 2000000 nanoseconds.</td>
</tr>
<tr>
<td>Step 4 (Optional) hardware profile latency monitor threshold-max &lt;value&gt;</td>
<td>Sets the maximum threshold for syslog generation. Range is 300 to 2000000 nanoseconds.</td>
</tr>
<tr>
<td>Step 5 (Optional) hardware profile latency monitor sampling &lt;value&gt;</td>
<td>Sets the sampling interval in seconds. Range is 1 to 30 seconds.</td>
</tr>
<tr>
<td>Step 6 exit</td>
<td>Updates the configuration and exits the configuration mode.</td>
</tr>
<tr>
<td>Step 7 (Optional) show hardware profile latency monitor summary</td>
<td>Displays the latency values on the packets.</td>
</tr>
</tbody>
</table>
Show Examples for Active Latency Monitoring

See the following examples that provide a real-time view of the latency incurred by the packets:

```
switch# show hardware profile latency monitor summary
10/13/2015 06:55:58
Device instance 0

Total Switch
------------
  3s  30s  1hr  All Time
Min Latency (ns)  390  375  n/a  369
Max Latency (ns)  775  1844  n/a  1950
Avg Latency (ns)  612  721  n/a  754
Std Deviation    205.24 117.23  n/a  69.17

Ethernet1/1
----------
  3s  30s  1hr  All Time
Min Latency (ns)  775  762  n/a  762
Max Latency (ns)  775  1757  n/a  1950
Avg Latency (ns)  775  838  n/a  870
Std Deviation    n/a  83.87  n/a  100.93

Ethernet1/13
-----------
  3s  30s  1hr  All Time
Min Latency (ns)  671  646  n/a  644
Max Latency (ns)  671  1844  n/a  1844
Avg Latency (ns)  671  736  n/a  740
Std Deviation    n/a  100.16  n/a  93.76
```
Configuring Active Latency Monitoring

Show Examples for Active Latency Monitoring
Link Level Flow Control

Link-level flow control is a congestion management technique that pauses data transmission until the congestion in the system is resolved. When a receiving device becomes congested, it communicates with the transmitter by sending a PAUSE frame. When the transmitting device receives a Pause frame it stops the transmission of any further data frames for a short period of time. The link-level flow control feature applies to all the traffic on the link. The transmit and receive directions are separately configurable. By default, link-level flow control is disabled for both directions.

Guidelines and Limitations for Link Level Flow Control

- Ethernet interfaces do not auto-detect the link-level flow control capability. You must configure the capability explicitly.
- When both link-level flow control (LLFC) and priority flow control (PFC) are enabled, LLFC takes precedence.
- Enabling link level flow control requires a part of the buffer to be reserved. This reduces the available shared buffer space.
- Flow control is not supported on 40G ports.
- Data Center Bridging Exchange Protocol (DCBX) is not supported.
- Configuration time quanta of the pause frames is not supported.
- Setting of pause threshold values is restricted.
• Configuring Link Level Flow Control on the interfaces will flap the interfaces which results in a momentary traffic loss.

• When a no-drop QoS group is configured, you must ensure that packets received on ports that do not have flow control send-on configured are not classified to a no-drop QoS group.

• Only a no-drop QoS group is capable of generating link level pause frames.

• It is recommended to use default buffer sizes for no-drop classes because if the buffer size is specified through CLI, it will allocate the same buffer size for all ports irrespective of the link speed, and MTU size.

• It is recommended to change the LLFC configuration when there is no traffic, otherwise packets already in the MMU of the system may not get the expected treatment.

• When configuring a no-drop class for QoS you must use QoS-Group 1 and map the QoS Group 1 to the no-drop class.

### Information About Link Level Flow Control

#### Link Level Flow Control on Interfaces

When link level flow control is configured the system changes the interface state to Down if the specified interface is in UP state and then applies the flow control configuration. After the configuration is successfully applied to the interface, the system restores the interface to the UP state.

#### Link Level Flow Control on Ports

During a port shutdown event, the flow-control settings on an interface are retained, however no traffic is received or transmitted on the link. During a port startup event the flow-control settings are reinstated on to the hardware.

#### Mismatched Link Level Flow Control Configurations

The transmit and receive directions can be configured separately, and each device on the network can have a different Link Level Flow Control (LLFC) configuration. The following table describes how devices with mis-matched configurations interact.

<table>
<thead>
<tr>
<th>Switch A</th>
<th>Switch B</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLFC configured to receive and transmit PAUSE frames.</td>
<td>LLFC configured to receive PAUSE frames.</td>
<td>Switch A can transmit 802.3x PAUSE frames and honor 802.3x PAUSE frames.</td>
</tr>
</tbody>
</table>
### Switches and Description

<table>
<thead>
<tr>
<th>Switch A</th>
<th>Switch B</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLFC configured to receive and transmit PAUSE frames.</td>
<td>LLFC configured to transmit PAUSE frames.</td>
<td>Switch A can transmit 802.3x PAUSE frames and honor 802.3x PAUSE frames. Switch B can transmit 802.3x PAUSE frames but will drop all received PAUSE frames.</td>
</tr>
</tbody>
</table>

### How to Configure Link Level Flow Control

#### Configuring Link Level Flow Control Receive

**SUMMARY STEPS**

1. enable
2. configure terminal
3. interface ethernet 1/1
4. flowcontrol receive on
5. exit

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> interface ethernet 1/1</td>
<td>Configures an interface type and enters interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# interface ethernet 1/1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> flowcontrol receive on</td>
<td>Enables the interface to receive and process pause frames.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-if)# flowcontrol receive on</td>
<td></td>
</tr>
</tbody>
</table>
### Purpose

**Command or Action**

**Purpose**

<table>
<thead>
<tr>
<th>Step 5</th>
<th>exit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong> Device(config-if)# exit</td>
<td></td>
</tr>
<tr>
<td>Exits interface configuration mode.</td>
<td></td>
</tr>
</tbody>
</table>

---

## Configuring Link Level Flow Control Transmit

To configure link-level flow control transmit on an interface, you enable flow control on the interface, configure a network-qos type QoS policy to enable a no-drop QoS group, and apply a qos type QoS policy to classify the traffic that requires no-drop behavior to the no-drop class.

You must ensure that bandwidth is allocated for the No-Drop QoS class using a queuing policy when you define a no-drop class. For more information, see the "Configuring Type Queuing Policies" section.

---

**Note**

When a no-drop QoS Group is configured you must ensure that packets received on ports that do not have flow-control send-on configured, are not classified to a no-drop QoS group. This is required as any ingress port that does not have flow-control send-on configured, can not generate a link level pause frame and there is no way to request the transmitting device to stop the transmission. Therefore, if flow-control send-on is not configured on all the interfaces you should not use a system policy to classify the packets to the no-drop QoS group. Instead, you should apply an interface QoS policy to the interfaces that having flow-control send-on enabled.

---

### SUMMARY STEPS

1. enable
2. configure terminal
3. interface ethernet 1/1
4. flowcontrol send on
5. exit
6. class-map type network-qos class-name
7. match qos-group group-number
8. policy-map type network-qos policy-map-name
9. class type network-qos class-name
10. pause no-drop
11. system qos
12. service-policy type network-qos policy-name
13. exit
14. show running ipqos

---

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>• Enter your password if prompted.</td>
</tr>
</tbody>
</table>

---

Cisco Nexus 3548 Switch NX-OS Quality of Service Configuration Guide, Release 9.2(x)
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device&gt; enable</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td>Configures an interface type and enters interface configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>interface ethernet 1/1</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# interface ethernet 1/1</td>
<td>Enables the interface to send pause frames to remote devices.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>flowcontrol send on</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-if)# flowcontrol send on</td>
<td>Exits interface configuration mode and returns to global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td></td>
</tr>
<tr>
<td>exit</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-if)# exit</td>
<td>Creates a network-qos class, and places the device in network-qos class-map configuration mode.</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td></td>
</tr>
<tr>
<td>class-map type network-qos class-name</td>
<td>Specifies the qos-group on which LLFC pause no-drop needs to be enabled.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# class-map type network-qos class1</td>
<td>Creates a network-qos policy map, and places the device in network-qos policy-map configuration mode.</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td></td>
</tr>
<tr>
<td>match qos-group group-number</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-cmap-nq)# match qos-group 1</td>
<td>Specifies the network-qos class map to use for matching, for this policy, and and places the device in network-qos policy-map-class configuration mode.</td>
</tr>
<tr>
<td><strong>Step 8</strong></td>
<td></td>
</tr>
<tr>
<td>policy-map type network-qos policy-map-name</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-cmap-nq)# policy-map type network-qos my_network_policy</td>
<td>Specifies the pause characteristics for this class.</td>
</tr>
</tbody>
</table>
### Command or Action
Device(config-pmap-nq-c)# pause no-drop

### Purpose
Enters QoS system configuration mode.

### Step 11
**system qos**
**Example:**
Device(config-pmap-nq-c)# system qos

**Step 12**
**service-policy type network-qos policy-name**
**Example:**
Device(config-sys-qos)# service-policy type network-qos my_network_policy

**Step 13**
**exit**
**Example:**
Device(config-sys-qos)# exit

**Step 14**
**show running ipqos**
**Example:**
Device# show running ipqos

---

### Configuration Examples for Link Level Flow Control

#### Example: Configuring Link Level Flow Control Receive

**Configuring Link Level Flow Control Receive**

The following example shows how to configure Link Level Flow Control receive on the device:

```
Device# configure terminal
Device(config)# interface ethernet 1/1
Device(config-if)# flowcontrol receive on
Device(config-if)# exit
```
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 is similar to 802.3x Flow Control (pause frames) or link-level flow control (LFC). However, PFC functions on a per class-of-service (CoS) basis.

When a buffer threshold is exceeded due to congestion, LFC sends a pause frame to its peer to pause all data transmission on the link for a specified period of time. When the congestion is mitigated (traffic comes under the configured threshold), a resume frame is generated to restart data transmission on the link.

In contrast, during 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.

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.

### Licensing Requirements for Priority Flow Control

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Product</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco NX-OS</td>
<td>The PFC feature does not require a license. Any feature not included in a license package is bundled with the NX-OS image and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the <a href="https://www.cisco.com/c/en/us/products/solutions/networking-nexus-3000-series-switches/nexus-3500-series-qual-of-service/configuration-guide-sg.html">Cisco NX-OS Licensing Guide</a>.</td>
</tr>
</tbody>
</table>

### Prerequisites for Priority Flow Control

PFC has the following prerequisites:

- You must be familiar with using modular QoS CLI.
- You are logged on to the device.

### Guidelines and Limitations for Priority Flow Control

PFC on the Nexus 3500 platform has the following configuration guidelines and limitations:

- PFC is supported only on the network-qos class matching qos-group 2 and qos-group 3. Configuring it on other qos-group matching classes results in an error.
- When configuring PFC in the network-qos policy, use the `pause pfc-cos` variant of the pause command.
- The `pause no-drop` variant of the pause command is used exclusively for configuring LLFC.
- We recommend not configuring LLFC and PFC together on the same switch. Choose only one of the following configurations for flow control:
  - **PFC Configuration**: only `pause pfc-cos` variant in network-qos config and enabling priority-flow-control on the port.
  - **LLFC Configuration**: only `pause no-drop` variant in network-qos policy and flow control send/receive on the port.
- For PFC to function correctly, the participating entities in a PFC enabled network should honor PFC frames, as per standard. When a peer honors PFC frames, PFC frames are generated only for the queue that is congested.
However, when a peer does not honor PFC frames, there is no provision to immediately drop the packets that have exceeded the buffer threshold. As a result, the entire no-drop buffer becomes exhausted and PFC frames are sent to other non-congested/no-traffic no-drop queues.

- PFC frames are generated for congested traffic that hit the multicast queue.
- The pause buffer threshold is configured in the network-qos policy.
- Adding pause buffer size threshold configuration is optional for cable lengths that are less than 100 meters and it need not be configured.
- For cable lengths that are greater than 100m, the pause buffer size threshold configuration is mandatory and it is required as part of the network QoS policy configuration.
- If PFC is enabled on a port or a port channel, it does not cause a port flap.
- PFC configuration enables PFC in both the send (Tx) and receive (Rx) direction.
- 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, Cisco recommends that you have only the no-drop class traffic on the queue.
- When a no-drop class is classified based on 802.1p CoS x and assigned an internal priority value (qos-group) of y, Cisco recommends that you use the internal priority value x to classify traffic on 802.1p CoS only, and not on any other field. The packet priority assigned is x if the classification is not based on CoS, which results in packets of internal priority x and y to map to the same priority x.
- The PFC feature supports up to two no-drop classes of any maximum transmission unit (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
  - Buffer-size of the pause threshold
  - Number of 10G and 40G ports
  - When configuring the pause threshold, with the default MTU and 10G ports, the maximum number of interfaces that can be enabled with PFC is approximately 20 interfaces.
- You can define the upper limit of any MTU in the system using the systemjumbomtu command. The MTU range is from 1500 to 9216 bytes, and the default is 9216 bytes.
- The interface QoS policy takes precedence over the system policy. PFC priority derivation also happens 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, Cisco recommends that you stop traffic before applying or removing a 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, Cisco recommends that you enable PFC on each interface through which the no-drop class traffic flows (Tx/Rx).

• Cisco recommends 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.

• Cisco recommends that you use default buffer sizes for no-drop classes or configure different network QoS policies suitable to 10G and 40G interfaces and the no-drop class MTU size. If the buffer size is specified through the CLI, it allocates the same buffer size for all ports irrespective of the link speed and MTU size. Applying the same pause buffer-size on 10G and 40G interfaces is not supported.

• Do not enable WRED on a no-drop class because it results in egress queue drops.

• For VLAN-tagged packets, priority is 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.

• For non VLAN-tagged frames, priority is assigned based on the set qos-group action given by the ingress QoS policy. Classification is based on a QoS policy-allowed match condition such as precedence, DSCP, or access-list. You must ensure that the pfc-cos value provided in the network-qos policy for this class is the same as the qos-group value in this case.

### Default Settings for Priority Flow Control

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFC</td>
<td>Auto</td>
</tr>
</tbody>
</table>

### 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 modes:

• on—Enables PFC on the local port regardless of the capability of the peers.

• off—Disables PFC on the local port.

### SUMMARY STEPS

1. configure terminal
2. interface type slot/port
3. priority-flow-control mode [ | off | on ]
4. show interface priority-flow-control
**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# configure terminal</td>
<td></td>
</tr>
<tr>
<td>switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>interface type slot/port</td>
<td>Enters interface mode on the interface specified.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config)# interface ethernet 2/5</td>
<td></td>
</tr>
<tr>
<td>switch(config-if)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>priority-flow-control mode [</td>
<td>off</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch(config-if)# priority-flow-control mode on</td>
<td></td>
</tr>
<tr>
<td>switch(config-if)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>show interface priority-flow-control</td>
<td>(Optional) Displays the status of PFC on all interfaces.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>switch# show interface priority-flow-control</td>
<td></td>
</tr>
</tbody>
</table>

**Enabling Priority Flow Control on a Traffic Class**

You can enable PFC on a particular traffic class.

**SUMMARY STEPS**

1. configure terminal
2. class-map type qos *class-name*
3. match cos *cos-value*
4. exit
5. policy-map type qos *policy-name*
6. class type qos *class-name*
7. set qos-group *qos-group-value*
8. exit
9. exit
10. class-map type network-qos match-any *class-name*
11. match qos-group *qos-group-value*
12. exit
13. policy-map type network-qos *policy-name*
14. class type network-qos *class-name*
15. pause pfc cos-value
16. exit
17. exit
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config)# configure terminal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch(config)#</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>class-map type qos class-name</td>
<td>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.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config)# class-map type qos c1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch(config-cmap-qos)#</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>match cos cos-value</td>
<td>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.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config-cmap-qos)# match cos 2</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>exit</td>
<td>Exits class-map mode and enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config-cmap-qos)# exit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch(config)#</td>
<td></td>
</tr>
<tr>
<td>Step 5</td>
<td>policy-map type qos policy-name</td>
<td>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.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config)# policy-map type qos p1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch(config-pmap-qos)#</td>
<td></td>
</tr>
<tr>
<td>Step 6</td>
<td>class type qos class-name</td>
<td>Associates a class map with the policy map and enters the configuration mode for the specified system class.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config-pmap-qos)# class type qos c1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch(config-pmap-c-qos)#</td>
<td><strong>Note</strong> The associated class map must be the same type as the policy map type.</td>
</tr>
<tr>
<td>Step 7</td>
<td>set qos-group qos-group-value</td>
<td>Configures one or more qos-group values to match on for classification of traffic into this class map. There is no default value.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config-pmap-c-qos)# set qos-group 2</td>
<td></td>
</tr>
<tr>
<td>Step 8</td>
<td>exit</td>
<td>Exits the system class configuration mode and enters policy-map mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>switch(config-pmap-c-qos)# exit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>switch(config-pmap-qos)#</td>
<td></td>
</tr>
<tr>
<td>Step 9</td>
<td>exit</td>
<td>Exits policy-map mode and enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
<td></td>
</tr>
</tbody>
</table>
| switch(config-pmap-qos)# exit  
switch(config)# | Switches to the global configuration mode. |
| **Step 10** | **class-map type network-qos match-any class-name**  
**Example:**  
switch(config)# class-map type network-qos match-any c1  
switch(config-cmap-nqos)# | Creates a named object that represents a class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, and are case sensitive, and can be up to 40 characters. |
| **Step 11** | **match qos-group qos-group-value**  
**Example:**  
switch(config-cmap-nqos)# match qos-group 2 | Configures the traffic class by matching packets based on a list of QoS group values. Supported on QoS group 2 and QoS group 3. |
| **Step 12** | **exit**  
**Example:**  
switch(config-cmap-nqos)# exit  
switch(config)# | Exits class-map mode and enters global configuration mode. |
| **Step 13** | **policy-map type network-qos policy-name**  
**Example:**  
switch(config)# policy-map type network-qos pl  
switch(config-pmap-nqos)# | 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, and are case sensitive, and can be up to 40 characters. |
| **Step 14** | **class type network-qos class-name**  
**Example:**  
switch(config-pmap-nqos)# class type network-qos c1  
switch(config-pmap-nqos-c)# | Associates a class map with the policy map, and enters the configuration mode for the specified system class.  
**Note** The associated class map must be the same type as the policy map type. |
| **Step 15** | **pause pfc cos-value**  
**Example:**  
switch(config-pmap-nqos-c)# pause pfc-cos 2 | PFC sends a pause frame that indicates which CoS value needs to be paused. (CoS value range is 0-7.)  
**Note** Nexus 3500 does not support the receive option of the pause command. An error occurs when the receive option is configured. |
| **Step 16** | **exit**  
**Example:**  
switch(config-pmap-nqos-c)# exit  
switch(config-pmap-nqos)# | Exits configuration mode and enters policy-map mode. |
| **Step 17** | **exit**  
**Example:**  
switch(config-pmap-nqos)# exit  
switch(config)# | Exits policy-map mode and enters global configuration mode. |
| **Step 18** | **system qos**  
**Example:** | Enters system class configuration mode. |
Purpose
Command or Action

switch(config)# system qos
switch(config-sys-qos)#

Step 19
service-policy type network-qos policy-name
Example:
switch(config-sys-qos)# service-policy type network-qos pl

Applies the policy map of type network-qos at the system level or to the specific interface.

Configuring Pause Buffer Thresholds

Pause buffer thresholds are configured in the network-qos policy. It is shared by all the ports in the system.

Note
Configuring pause thresholds in the ingress queuing policy is not supported for the Nexus 3500.

SUMMARY STEPS

1. configure terminal
2. policy-map type queuing policy-map-name
3. class-map type network-qos class-map-name
4. pause buffer-size buffer-size pause threshold xoff-size resume threshold xon-size pfc-cos pfc-cos-value
5. no pause buffer-size buffer-size pause threshold xoff-size resume-threshold xon-size pfc-cos pfc-cos-value

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Step 2 policy-map type queuing policy-map-name</td>
<td>Enters policy-map queuing class mode and identifies the policy map assigned to the type queuing policy map.</td>
</tr>
<tr>
<td>Step 3 class-map type network-qos class-map-name</td>
<td>Attaches the class map of type network-qos and then enters class network-qos mode.</td>
</tr>
<tr>
<td>Step 5 no pause buffer-size buffer-size pause threshold xoff-size resume-threshold xon-size pfc-cos pfc-cos-value</td>
<td>Removes the buffer threshold settings for pause and resume.</td>
</tr>
</tbody>
</table>

Example

Example:

switch(config-cmap-nqos)# class type network-qos nc2
Configuring Queue Limits

The queue-limit is configured with the network-qos policy.

Note

A queue-limit is configurable under a no-drop (PFC) enabled network-qos class. However, a queue-limit is not effective on such classes.

**SUMMARY STEPS**

1. configure terminal
2. policy-map type network-qos policy-map-name
3. class-map type network-qos class-map-name
4. queue-limit queue-size bytes

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure terminal</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>policy-map type network-qos policy-map-name</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>class-map type network-qos class-map-name</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>queue-limit queue-size bytes</td>
</tr>
</tbody>
</table>

**Verifying the Priority Flow Control Configuration**

To display the PFC configuration, perform the following task:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show interface priority-flow-control [module [number]]</td>
<td>Displays the status of PFC on all interfaces or on specific modules.</td>
</tr>
<tr>
<td>show interface priority-flow-control [detail] module [number]</td>
<td>Displays the detail status of PFC on all interfaces or on specific modules.</td>
</tr>
</tbody>
</table>
Configuration Examples for Priority Flow Control

The following example shows how to configure PFC:

```
configure terminal
interface ethernet 1/1
priority-flow-control mode on
```

The following example shows how to enable PFC on a traffic class:

```
switch(config)# class-map type qos c2
switch(config-cmap-qos)# match cos 2
switch(config-cmap-qos)# exit

switch(config)# policy-map type qos p1
switch(config-pmap-qos)# class type qos c2
switch(config-pmap-c-qos)# set qos-group 2
switch(config-pmap-c-qos)# exit
switch(config-pmap-qos)# exit

switch(config)# class-map type queuing cq2
switch(config-cmap-que)# match qos-group 2
switch(config-cmap-que)# exit

switch(config)# policy-map type queuing pql
switch(config-pmap-que)# class type queuing cq2
switch(config-pmap-c-que)# bandwidth percent 20
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# exit

switch(config)# class-map type network-qos cn1
switch(config)# class-map type network-qos n2
switch(config-cmap-nqos)# match qos-group 2
switch(config-cmap-nqos)# exit

switch(config)# policy-map type network-qos pn1
switch(config-pmap-nqos)# class type network-qos n2
switch(config-pmap-nqos-c)# pause pfc-cos 2
switch(config-pmap-nqos-c)# exit
switch(config-pmap-nqos)# exit

switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos pn1
switch(config-sys-qos)# service-policy type qos input p1
switch(config-sys-qos)# service-policy type queuing output pql
```
When there is an issue with priority-flow-control functionality, collect the output from the following commands for troubleshooting:

- **show tech-support module 1**
  - Displays internal QoS hardware buffer/configuration information.

- **show tech-support aclqos**
  - Displays PFC configuration/status commands.

- **show tech-support**
  - Displays `show running config` output along with other QoS internal commands.