Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide, Release 6.x

First Published: 2013-11-20

Last Modified: 2020-04-22

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

CONTENTS

Preface

This preface includes the following sections:

Audience

This publication is for network administrators who install, configure, and maintain Cisco Nexus switches.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
Italic	Italic text indicates arguments for which you supply the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x | y]	Square brackets enclosing keywords or arguments that are separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments that are separated by a vertical bar indicate a required choice.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.

Convention	Description
`variable`	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string includes the quotation marks.

Examples use the following conventions:

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

Related Documentation for Cisco Nexus 9000 Series Switches

The entire Cisco Nexus 9000 Series switch documentation set is available at the following URL:

http://www.cisco.com/en/US/products/ps13386/tsd_products_support_series_home.html

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus9k-docfeedback@cisco.com. We appreciate your feedback.

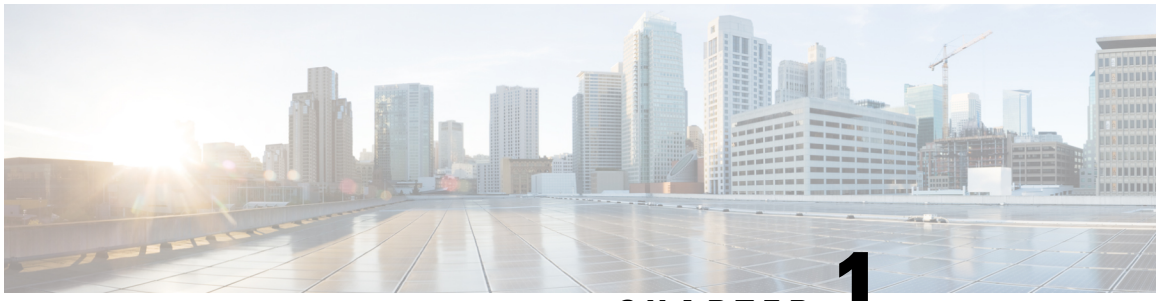
Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Preface



CHAPTER 1

New and Changed Information

This chapter provides release-specific information for each new and changed feature in the *Cisco Nexus 9000 Series NX-OS QoS Configuration Guide*.

- New and Changed Information, on page 1

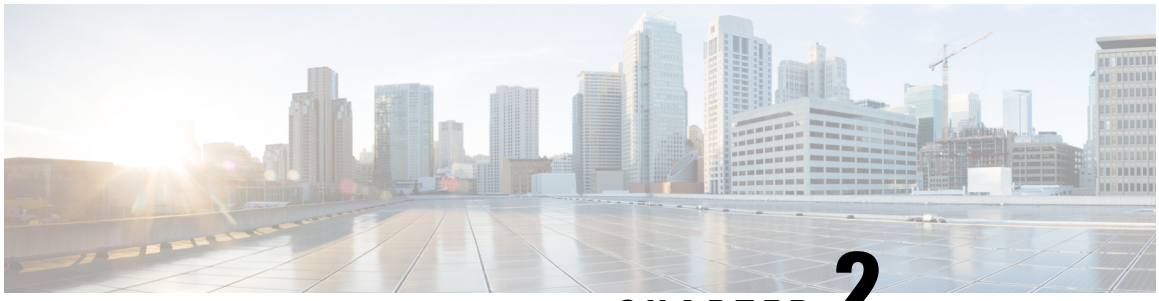
New and Changed Information

This table summarizes the new and changed features for the *Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide* and where they are documented.

Table 1: New and Changed Features

Feature	Description	Changed in Release	Where Documented
FEX QoS policy	Added support for the classification of traffic classes and the marking of incoming and outgoing packets.	6.1(2)I3(2)	Configuring Marking Configuring Classification
QoS TCAM Lite Regions	Enables configuration of single wide QoS TCAM entries for IPV4. QoS TCAM lite regions support QoS policies for Layer 2, Layer 3, VLAN, and FEX interfaces.	6.1(2)I3(2)	About QoS TCAM Lite Regions
Link Level Flow	Added support for link level flow control.	6.1(2)I3(1)	Link Level Flow Control
Buffer Optimization	Added support for controlling the QoS shared buffer.	6.1(2)I3(1)	Explicit Congestion Notification
Ingress Queuing	Added support for the ingress queuing policy.	6.1(2)I3(1)	Configuring Pause Buffer Thresholds and Queue Limit Using Ingress Queuing Policy

Feature	Description	Changed in Release	Where Documented
8q Mode	Added support for system-defined objects for the 8q mode feature.	6.1(2)I1(3)	System-Defined MQC Objects
8q Mode	Added support for system-defined objects for the 8q mode feature.	6.1(2)I2(2a)	System-Defined MQC Objects
AFD	Added support for approximate fair-drop (AFD) feature.	6.1(2)I2(2)	Explicit Congestion Notification
Buffer-boost	Added support for enabling extra buffers with the buffer-boost feature .	6.1(2)I2(1)	Guidelines and Limitations
TCAM carving	Added support for changing the size of the access control list (ACL) with the ternary content addressable memory (TCAM) feature.	6.1(2)I2(1)	About QoS TCAM Carving



CHAPTER 2

Overview

Licensing Requirements

For a complete explanation of Cisco NX-OS licensing recommendations and how to obtain and apply licenses, see the *Cisco NX-OS Licensing Guide*.

About 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, police and prioritize the traffic flow, and help avoid traffic congestion in a network. The control of traffic is based on the fields in the packets that flow through the system. You use the Modular QoS (MQC) CLI to create the traffic classes and policies of the QoS features.

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

- QoS policies include classification and marking features.
- QoS policies include policing features.
- QoS policies include shaping, weighted random early detection (WRED), and explicit congestion notification (ECN) features.

- Queuing policies use the queuing and scheduling features.

Note The system-defined QoS features and values that are discussed in the "Using Modular QoS CLI" section apply globally to the entire device and cannot be modified.

Using QoS

Traffic is processed based on how you classify it and the policies that you create and apply to traffic classes.

To configure QoS features, you use the following steps:

1. Create traffic classes by classifying the incoming packets that match criteria such as IP address or QoS fields.
2. Create policies by specifying actions to take on the traffic classes, such as policing, marking, or dropping packets.
3. Apply policies to a port, port channel, or subinterface.

You use MQC to create the traffic classes and policies of the QoS features.

Note The queuing and scheduling operations of the overall QoS feature are applicable to both IPv4 and IPv6.

Note IP tunnels do not support access control lists (ACLs) or QoS policies.

Classification

You use classification to partition traffic into classes. You classify the traffic based on the port characteristics or the packet header fields that include IP precedence, differentiated services code point (DSCP), Layer 3 to Layer 4 parameters, and the packet length.

The values used to classify traffic are called match criteria. When you define a traffic class, you can specify multiple match criteria, you can choose to not match on a particular criterion, or you can determine the traffic class by matching any or all criteria.

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

Marking

Marking is the setting of QoS information that is related to a packet. You can set the value of a standard QoS field for COS, IP precedence and DSCP, and internal labels (such as QoS groups) that can be used in subsequent actions. Marking QoS groups is used to identify the traffic type for queuing and scheduling traffic.

Policing

Policing is the monitoring of data rates for a particular class of traffic. The device can also monitor associated burst sizes.

Single-rate policers monitor the specified committed information rate (CIR) of traffic. Dual-rate policers monitor both CIR and peak information rate (PIR) of traffic.

Queuing and Scheduling

The queuing and scheduling process allows you to control the bandwidth allocated to traffic classes so that you achieve the desired trade-off between throughput and latency.

You can apply weighted random early detection (WRED) to a class of traffic, which allows packets to be dropped based on the QoS group. The WRED algorithm allows you to perform proactive queue management to avoid traffic congestion.

You can shape traffic by imposing a maximum data rate on a class of traffic so that excess packets are retained in a queue to smooth (constrain) the output rate. In addition, minimum bandwidth shaping can be configured to provide a minimum guaranteed bandwidth for a class of traffic.

You can limit the size of the queues for a particular class of traffic by applying either static or dynamic limits.

ECN can be enabled along with WRED on a particular class of traffic to mark the congestion state instead of dropping the packets.

Sequencing of QoS Actions

The following are the three types of policies:

- **network qos**—Defines the characteristics of QoS properties network wide.
- **qos**—Defines MQC objects that you can use for marking and policing.
- **queuing**—Defines MQC objects that you can use for queuing and scheduling.

Note The default type of policy is **qos**.

The system performs actions for QoS policies only if you define them under the type qos service policies.

Sequencing of Ingress Traffic Actions

The sequence of QoS actions on ingress traffic is as follows:

1. Classification
2. Marking
3. Policing

Sequencing of Egress Traffic Actions

The sequencing of QoS actions on egress traffic is as follows:

1. Queuing and scheduling

High Availability Requirements for QoS Features

The Cisco NX-OS QoS software recovers its previous state after a software restart, and it is capable of a switchover from the active supervisor to the standby supervisor without a loss of state.

Note For complete information on high availability, see the *Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*.

QoS Feature Configuration with MQC

You use MQC to configure QoS features. The MQC configuration commands are shown in the following table:

Table 2: MQC Configuration Commands

MQC Command	Description
class-map	Defines a class map that represents a class of traffic.
policy-map	Defines a policy map that represents a set of policies to be applied to a set of class maps.

You can modify or delete MQC objects, except system-defined objects, when the objects are not associated with any interfaces.

After a QoS policy is defined, you can attach the policy map to an interface by using the interface configuration command shown in the following table:

Table 3: Interface Command to Attach a Policy Map to an Interface

Interface Command	Description
service-policy	Applies the specified policy map to input or output packets on the interface.

QoS Statistics

Statistics are maintained for each policy, class action, and match criteria per interface. You can enable or disable the collection of statistics, you can display statistics using the **show policy-map** interface command,

and you can clear statistics based on an interface or policy map with the **clear qos statistics** command. Statistics are enabled by default and can be disabled globally.

Default QoS Behavior

The QoS queuing features are enabled by default. Specific QoS-type features, such as policing and marking, are enabled only when a policy is attached to an interface. Specific policies are enabled when that policy is attached to an interface.

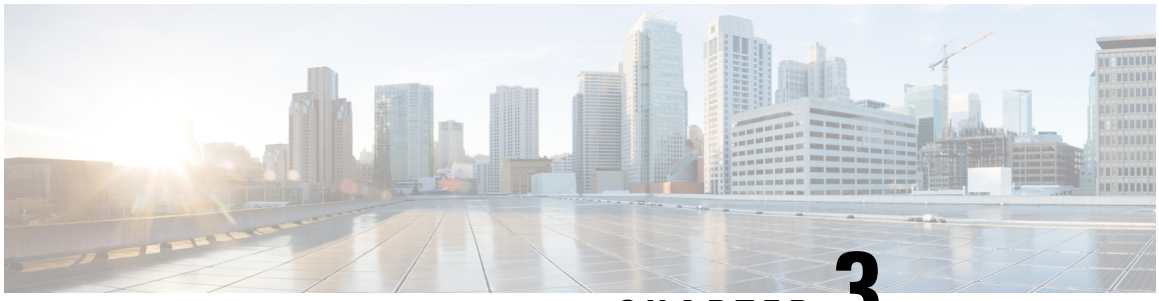
By default, the device always enables a system default queuing policy, or system-defined queuing policy map, on each port and port channel. When you configure a queuing policy and apply the new queuing policy to specified interfaces, the new queuing policy replaces the default queuing policy, and those rules now apply.

The device enables other QoS features, policing and marking, only when you apply a policy map to an interface.

Virtual Device Contexts

Cisco NX-OS can segment operating system and hardware resources into virtual device contexts (VDCs) that emulate virtual devices. The Cisco Nexus 9000 Series device currently does not support multiple VDCs. All device resources are managed in the default VDC.

Note The VDC feature is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).



CHAPTER 3

Using Modular QoS CLI

About MQC

Cisco Modular Quality of Service Command Line Interface (MQC) provides a language to define QoS policies.

You configure QoS policies by following these three steps:

1. Define traffic classes.
2. Associate policies and actions with each traffic class.
3. Attach policies to logical or physical interfaces.

MQC provides a command type to define traffic classes and policies:

- **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 marking and policing.
- **queuing**—Defines MQC objects that you can use for queuing and scheduling.

Note The **qos** type is the default.

Egress QoS policies are not supported on the subinterfaces.

You can attach policies to ports, port channels, or subinterfaces 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.

Caution In the interface configuration mode, the device can accept QoS and access control list (ACL) commands irrespective of the line card on which the interface host is up or down. However, you cannot enter the interface submode when the line card is down because the device does not accept any preconfiguration information.

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 device 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 device, and their attributes.

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

When you configure QoS features, and the system requests MQC objects, you can use system-defined MQC objects for 4q mode or system-defined objects for 8q mode.

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

Default System Classes

The device provides the following system classes:

- Drop system class

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

Using an MQC Object

You configure QoS and queuing policies using the MQC class-map and policy-map objects. After you configure class maps and policy maps, you can attach one policy map of each type to an interface. A QoS policy can only be applied to the ingress direction.

A policy map contains either a QoS policy or queuing policy. The policy map references the names of class maps that represent traffic classes. For each class of traffic, the device applies the policies on the interface or VLAN that you select.

A packet is matched sequentially to a class of traffic starting from the first traffic class definition. When a match is found, the policy actions for that class are applied to the packet.

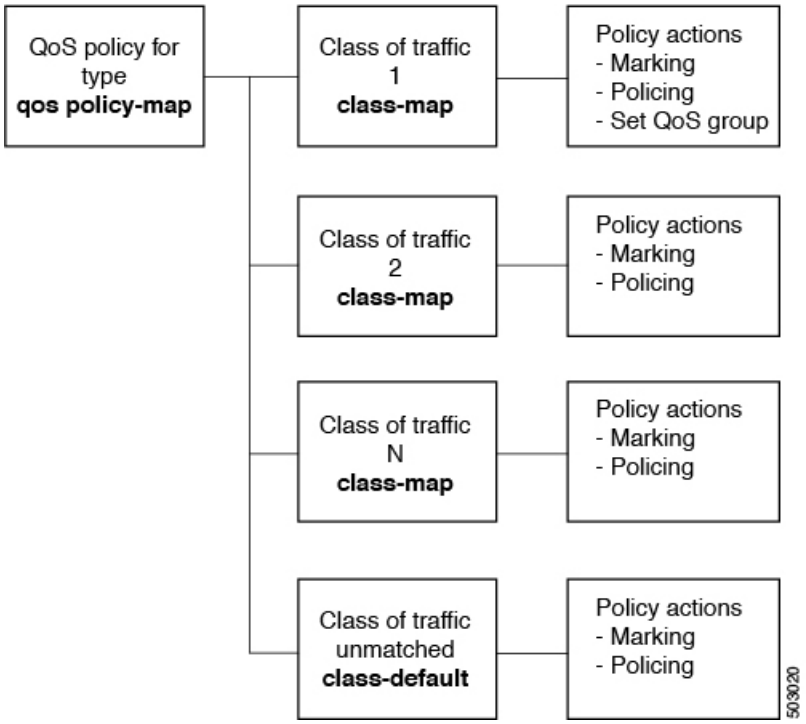
The reserved class map class-default receives all traffic that is not matched in type qos policies, and the device applies the policy actions as it would for any other traffic class.

Type qos Policies

You use type qos policies to mark and to police packets, and to set qos-groups, which drive matching conditions for system-defined type network-qos and type queuing class-maps.

The following figure shows the QoS policy structure with the associated MQC objects of type qos. The MQC objects are shown in bold.

Figure 1: QoS Policy Diagram Showing Type qos MQC Object Usage

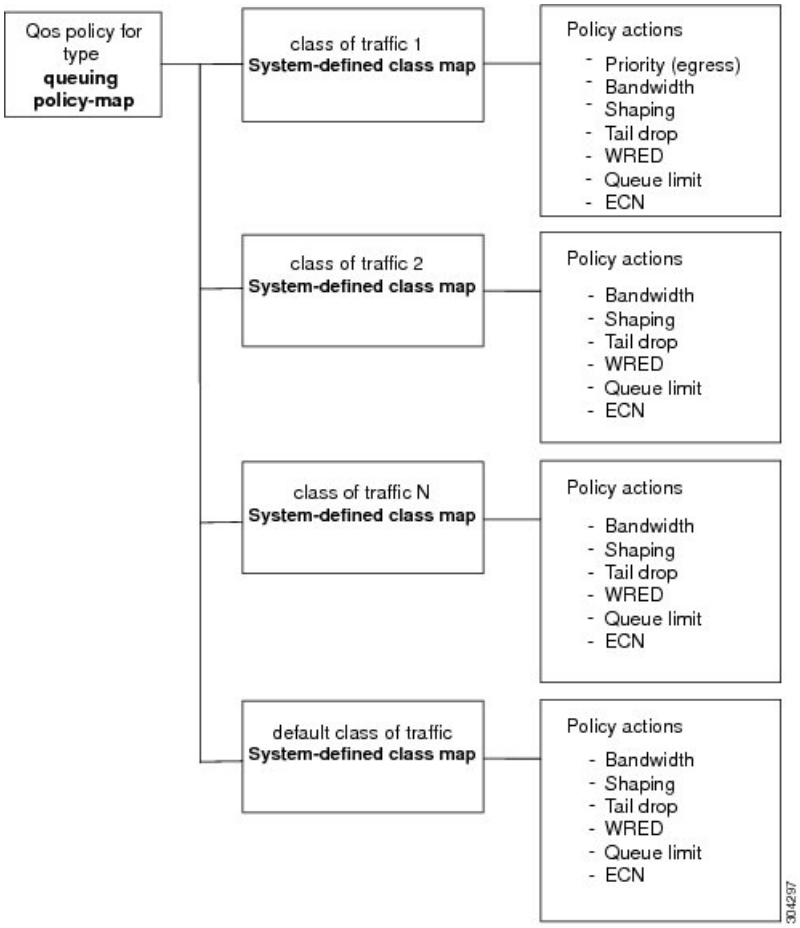


Type Queuing Policies

You use type queuing policies to shape and queue packets.

The following figure shows the QoS policy structure with associated MQC objects of type queuing. The MQC objects are shown in bold.

Figure 2: QoS Policy Diagram Showing Type Queuing MQC Object Usage



Note: See the "Configuring Queuing and Scheduling" chapter for information on configuring these parameters.

System-Defined MQC Objects

When you configure QoS features, and the system requests MQC objects, you can use system-defined objects for 4q mode or system-defined objects for 8q mode.

The system-defined objects for 8q mode are supported on the following devices:

- N9K-C92348GC-X
- Cisco Nexus 9300-EX switches
- Cisco Nexus 9300-FX switches
- Cisco Nexus 9300-FX2 switches
- Cisco Nexus 9300-GX switches
- Cisco Nexus 9504, 9508, and 9516 switches with -EX or -FX line cards.

Note When FEX is connected, it should be configured with 4q.

Note The following Cisco Nexus switches and line cards do not support system-defined objects for 8q mode:

- N9K-C9272Q
- N9K-C9332PQ
- N9K-C93120TX
- N9K-X9464PX
- N9K-X9432PQ

Note System-defined objects for 8q mode are not supported on ACI (Application Centric Infrastructure) capable linecards.

System-Defined MQC Objects for 4q Mode

When you configure QoS features, and the system requests MQC objects, you can use the following system-defined objects:

Note The Cisco Nexus 9000 series NX-OS system operates in 4q mode by default. System-defined MQC objects for 4q mode are the default MQC objects.

Note System-defined MQC objects for 4q mode are not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

- Type qos class maps

Table 4: System-Defined Type qos Class Maps

Class Map Name	Description
class-default	Type qos class map that is assigned to all packets that match none of the criteria of traffic classes that you define in a type qos policy map.

- Type queuing class maps

Table 5: System-Defined Type queuing Class Maps for 4q Mode

Class Map Queue Name	Description
c-out-q-default	Egress default queue — QoS group 0
c-out-q1	Egress queue 1 — QoS group 1
c-out-q2	Egress queue 2 — QoS group 2
c-out-q3	Egress queue 3 — QoS group 3

- Type network-qos class maps

Table 6: System-Defined Type network-qos Class Maps for 4q Mode

Class Map Network-QoS Name	Description
c-nq-default	Network-qos class — QoS group 0
c-nq1	Network-qos class — QoS group 1
c-nq2	Network-qos class — QoS group 2
c-nq3	Network-qos class — QoS group 3

- Policy maps

Table 7: System-Defined Queuing Policy Maps for 4q Mode

Queuing Policy Map Name	Description
default-out-policy	Output queuing policy map that is attached to all module ports to which you do not apply a queuing policy map. The default configuration values are as follows: `policy-map type queuing default-out-policy` `  class type queuing c-out-q3` `   priority level 1` `  class type queuing c-out-q2` `   bandwidth remaining percent 0` `  class type queuing c-out-q1` `   bandwidth remaining percent 0` `  class type queuing c-out-q-default` `   bandwidth remaining percent 100`

Queuing Policy Map Name	Description
default-network-qos-policy	Network-qos queuing policy map that is attached to all module ports to which you do not apply a queuing policy map. The default configuration values are as follows: `policy-map type network-qos default-nq-policy` `  class type network-qos c-nq3` `   match qos-group 3` `   mtu 1500` `  class type network-qos c-nq2` `   match qos-group 2` `   mtu 1500` `  class type network-qos c-nq1` `   match qos-group 1` `   mtu 1500` `  class type network-qos c-nq-default` `   match qos-group 0` `   mtu 1500`

System-Defined MQC Objects for 8q Mode

When you configure QoS features, and the system requests MQC objects, you can use the following system-defined objects:

Note System-defined MQC objects for 4q mode are the default MQC objects. You must enable the following MQC objects to change to 8q mode.

- Type qos class maps

Table 8: System-Defined Type qos Class Maps

Class Map Name	Description
class-default	Type qos class map that is assigned to all packets that match none of the criteria of traffic classes that you define in a type qos policy map.

- Type queuing class maps

Table 9: System-Defined Type queuing Class Maps for 8q Mode (Egress)

Class Map Queue Name	Description
c-out-8q-q-default	Egress default queue — QoS group 0
c-out-8q-q1	Egress queue 1 — QoS group 1
c-out-8q-q2	Egress queue 2 — QoS group 2
c-out-8q-q3	Egress queue 3 — QoS group 3
c-out-8q-q4	Egress queue 4 — QoS group 4

Class Map Queue Name	Description
c-out-8q-q5	Egress queue 5 — QoS group 5
c-out-8q-q6	Egress queue 6 — QoS group 6
c-out-8q-q7	Egress queue 7 — QoS group 7

Table 10: System-Defined Type queuing Class Maps for 8q Mode (Ingress)

Class Map Queue Name	Description
c-in-q-default	Ingress default queue — QoS group 0
c-in-q1	Ingress queue 1 — QoS group 1
c-in-q2	Ingress queue 2 — QoS group 2
c-in-q3	Ingress queue 3 — QoS group 3
c-in-q4	Ingress queue 4 — QoS group 4
c-in-q5	Ingress queue 5 — QoS group 5
c-in-q6	Ingress queue 6 — QoS group 6
c-in-q7	Ingress queue 7 — QoS group 7

- Type network-qos class maps

Note The System-Defined Type network-qos Class Maps for 8q Mode are not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

Table 11: System-Defined Type network-qos Class Maps for 8q Mode

Class Map Network-QoS Name	Description
c-8q-nq-default	Network-qos class — QoS group 0
c-8q-nq1	Network-qos class — QoS group 1
c-8q-nq2	Network-qos class — QoS group 2
c-8q-nq3	Network-qos class — QoS group 3
c-8q-nq4	Network-qos class — QoS group 4
c-8q-nq5	Network-qos class — QoS group 5
c-8q-nq6	Network-qos class — QoS group 6

Class Map Network-QoS Name	Description
c-8q-nq7	Network-qos class — QoS group 7

- Policy maps

Table 12: System-Defined Queuing Policy Maps for 8q Mode

Queuing Policy Map Name	Description
default-8q-out-policy	Output queuing policy map that is attached to all module ports to which you do not apply a queuing policy map. The default configuration values are as follows: `policy-map type queuing default-8q-out-policy` `  class type queuing c-out-8q-q7` `   priority level 1` `  class type queuing c-out-8q-q6` `   bandwidth remaining percent 0` `  class type queuing c-out-8q-q5` `   bandwidth remaining percent 0` `  class type queuing c-out-8q-q4` `   bandwidth remaining percent 0` `  class type queuing c-out-8q-q3` `   bandwidth remaining percent 0` `  class type queuing c-out-8q-q2` `   bandwidth remaining percent 0` `  class type queuing c-out-8q-q1` `   bandwidth remaining percent 0` `  class type queuing c-out-8q-q-default` `   bandwidth remaining percent 100`

Queuing Policy Map Name	Description
default-8q-network-qos-policy	Network-qos queuing policy map that is attached to all module ports to which you do not apply a queuing policy map. The default configuration values are as follows: `policy-map type network-qos default-8q-nq-policy` `  class type network-qos c-8q-nq7` `   match qos-group 7` `   mtu 1500` `  class type network-qos c-8q-nq6` `   match qos-group 6` `   mtu 1500` `  class type network-qos c-8q-nq5` `   match qos-group 5` `   mtu 1500` `  class type network-qos c-8q-nq4` `   match qos-group 4` `   mtu 1500` `  class type network-qos c-8q-nq3` `   match qos-group 3` `   mtu 1500` `  class type network-qos c-8q-nq2` `   match qos-group 2` `   mtu 1500` `  class type network-qos c-8q-nq1` `   match qos-group 1` `   mtu 1500` `  class type network-qos c-8q-nq-default` `   match qos-group 0` `   mtu 1500`

Changing to 8q Mode

Note The Cisco Nexus 9000 series NX-OS system operates in 4q mode by default.

Use the following guidelines to change to 8q mode:

- Change the network-qos policy to 8q mode.

  You can either activate the default-8q-nq-policy (which is the system created 8q default network-qos policy); or you can copy it using the **qos copy policy-map type network-qos** command, edit it as needed, and activate it.

- Change the queuing policy to 8q mode. (This means changing the system queuing policy and optionally any interface queuing policy.)

  Make a copy of the default-8q-out-policy (the default 8q queuing policy created by the system) using the **qos copy policy-map type queuing** command. Edit the copy of the default-8q-out-policy as needed and activate it at the system level and optionally at the interface level.

- After the network-qos and queuing policies are changed to 8q mode, you can start using **set qos-group** action for qos-groups 4-7 to steer the traffic to queues 4-7.

Notes About 8q Mode

The following are notes about 8q mode:

- When 8q policies are in active use, the system cannot be downgraded to a system image that does not support 8q mode.

  Note As a best practice to avoid incompatibilities, remove the 8q policies before a downgrade.

  The following example shows some incompatibilities when trying to downgrade to a system image that does not support 8q mode.

```
switch# show incompatibility nxos bootflash:n9000-dk9.6.1.2.I1.2.bin

The following configurations on active are incompatible with  the system image

1) Service : ipqosmgr , Capability : CAP_FEATURE_IPQOS_8Q_QUE_POLICY_ACTIVE
Description : QoS Manager - 8Q queuing policy active
Capability requirement : STRICT
Enable/Disable command : Please remove 8q queuing policy

2) Service : ipqosmgr , Capability : CAP_FEATURE_IPQOS_8Q_NQOS_POLICY_ACTIVE
Description : QoS Manager - 8Q network-qos policy active
Capability requirement : STRICT
Enable/Disable command : Please remove 8q network-qos policy
```

- No 8q policies can be activated on a system that has linecards that do not support 8-queues. All ACI (Application Centric Infrastructure) capable linecards do not support 8-queues.

  Note As a best practice, power off all linecards that do not support 8-queues before using 8-queue functionality.

  The following example shows some of the errors that occur when you attempt to use 8-queue functionality on a system that has linecards that do not support 8-queues.

```
switch(config)# system qos
  switch(config-sys-qos)# service-policy type queuing output default-8q-out-policy
  ERROR: policy-map default-8q-out-policy can be activated only on 8q capable platforms


  switch(config)# system qos
  switch(config-sys-qos)# service-policy type network-qos default-8q-nq-policy
  ERROR: policy-map default-8q-nq-policy can be activated only on 8q capable platforms


  switch(config)# policy-map p1
  switch(config-pmap-qos)# class c1
  switch(config-pmap-c-qos)# set qos-group 7
  ERROR: set on qos-group 4-7 is supported only on 8q capable platforms
```

Example of Changing to 8q Mode

The following is an example of changing to 8q mode:

Note This example is not applicable to the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

```
switch# qos copy policy-map type network-qos default-8q-nq-policy prefix my
switch# show policy-map type network-qos

  Type network-qos policy-maps
  ============================
  policy-map type network-qos my8q-nq
    class type network-qos c-8q-nq7
      mtu 1500
    class type network-qos c-8q-nq6
      mtu 1500
    class type network-qos c-8q-nq5
      mtu 1500
    class type network-qos c-8q-nq4
      mtu 1500
    class type network-qos c-8q-nq3
      mtu 1500
    class type network-qos c-8q-nq2
      mtu 1500
    class type network-qos c-8q-nq1
      mtu 1500
    class type network-qos c-8q-nq-default
      mtu 1500

switch# config t
switch(config)# policy-map type network-qos my8q-nq
switch(config-pmap-nqos)# class type network-qos c-8q-nq1
switch(config-pmap-nqos-c)# mtu 9216
switch(config-pmap-nqos-c)# class type network-qos c-8q-nq2
switch(config-pmap-nqos-c)# mtu 2240
switch(config-pmap-nqos-c)# class type network-qos c-8q-nq4
switch(config-pmap-nqos-c)# pause pfc-cos 4
switch(config-pmap-nqos-c)# class type network-qos c-8q-nq5
switch(config-pmap-nqos-c)# mtu 2240
switch(config-pmap-nqos-c)# pause pfc-cos 5
switch(config-pmap-nqos-c)# class type network-qos c-8q-nq6
switch(config-pmap-nqos-c)# mtu 9216
switch(config-pmap-nqos-c)# pause pfc-cos 6
switch(config-pmap-nqos-c)# show policy-map type network-qos my8q-nq

  Type network-qos policy-maps
  ============================
  policy-map type network-qos my8q-nq
    class type network-qos c-8q-nq7
      mtu 1500
    class type network-qos c-8q-nq6
      pause pfc-cos 6
      mtu 9216
    class type network-qos c-8q-nq5
      pause pfc-cos 5
      mtu 2240
    class type network-qos c-8q-nq4
      pause pfc-cos 4
      mtu 1500
    class type network-qos c-8q-nq3
      mtu 1500
    class type network-qos c-8q-nq2
      mtu 2240
    class type network-qos c-8q-nq1
```

```
      mtu 9216
    class type network-qos c-8q-nq-default
      mtu 1500

switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos my8q-nq
switch(config-sys-qos)# 2014 Jun 12 11:13:48 switch %$ VDC-1 %$
%IPQOSMGR-2-QOSMGR_NETWORK_QOS_POLICY_CHANGE: Policy my8q-nq is now active

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

  Type network-qos policy-maps
  ============================
  policy-map type network-qos my8q-nq
    class type network-qos c-8q-nq7
      match qos-group 7
      mtu 1500
    class type network-qos c-8q-nq6
      match qos-group 6
      pause pfc-cos 6
      mtu 9216
    class type network-qos c-8q-nq5
      match qos-group 5
      pause pfc-cos 5
      mtu 2240
    class type network-qos c-8q-nq4
      match qos-group 4
      pause pfc-cos 4
      mtu 1500
    class type network-qos c-8q-nq3
      match qos-group 3
      mtu 1500
    class type network-qos c-8q-nq2
      match qos-group 2
      mtu 2240
    class type network-qos c-8q-nq1
      match qos-group 1
      mtu 9216
    class type network-qos c-8q-nq-default
      match qos-group 0
      mtu 1500

switch# qos copy policy-map type queuing default-8q-out-policy prefix my
switch# show policy-map type queuing my8q-out


  Type queuing policy-maps
  ========================

  policy-map type queuing my8q-out
    class type queuing c-out-8q-q7
      priority level 1
    class type queuing c-out-8q-q6
      bandwidth remaining percent 0
    class type queuing c-out-8q-q5
      bandwidth remaining percent 0
    class type queuing c-out-8q-q4
      bandwidth remaining percent 0
    class type queuing c-out-8q-q3
      bandwidth remaining percent 0
    class type queuing c-out-8q-q2
      bandwidth remaining percent 0
    class type queuing c-out-8q-q1
      bandwidth remaining percent 0
```

```
    class type queuing c-out-8q-q-default
     bandwidth remaining percent 100

switch# config t
switch(config)# policy-map type queuing my8q-out
switch(config-pmap-c-que)# class type queuing c-out-8q-q-default
switch(config-pmap-c-que)# bandwidth remaining percent 30
switch(config-pmap-c-que)# class type queuing c-out-8q-q1
switch(config-pmap-c-que)# bandwidth remaining percent 15
switch(config-pmap-c-que)# class type queuing c-out-8q-q2
switch(config-pmap-c-que)# bandwidth remaining percent 15
switch(config-pmap-c-que)# class type queuing c-out-8q-q3
switch(config-pmap-c-que)# bandwidth remaining percent 10
switch(config-pmap-c-que)# class type queuing c-out-8q-q4
switch(config-pmap-c-que)# bandwidth remaining percent 10
switch(config-pmap-c-que)# class type queuing c-out-8q-q5
switch(config-pmap-c-que)# bandwidth remaining percent 10
switch(config-pmap-c-que)# class type queuing c-out-8q-q6
switch(config-pmap-c-que)# bandwidth remaining percent 10
switch(config-pmap-c-que)# show policy-map type queuing my8q-out


  Type queuing policy-maps
  ========================

  policy-map type queuing my8q-out
    class type queuing c-out-8q-q7
      priority level 1
    class type queuing c-out-8q-q6
      bandwidth remaining percent 10
    class type queuing c-out-8q-q5
      bandwidth remaining percent 10
    class type queuing c-out-8q-q4
      bandwidth remaining percent 10
    class type queuing c-out-8q-q3
      bandwidth remaining percent 10
    class type queuing c-out-8q-q2
      bandwidth remaining percent 15
    class type queuing c-out-8q-q1
      bandwidth remaining percent 15
    class type queuing c-out-8q-q-default
      bandwidth remaining percent 30

switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing output my8q-out
switch(config-sys-qos)# show policy-map system type queuing



  Service-policy output:   my8q-out
          Service-policy (queuing) output:   my8q-out
            policy statistics status:   disabled (current status: disabled)

            Class-map (queuing):   c-out-8q-q7 (match-any)
              priority level 1

            Class-map (queuing):   c-out-8q-q6 (match-any)
              bandwidth remaining percent 10

            Class-map (queuing):   c-out-8q-q5 (match-any)
              bandwidth remaining percent 10

            Class-map (queuing):   c-out-8q-q4 (match-any)
              bandwidth remaining percent 10
```

```
Class-map (queuing):   c-out-8q-q3 (match-any)
  bandwidth remaining percent 10

Class-map (queuing):   c-out-8q-q2 (match-any)
  bandwidth remaining percent 15

Class-map (queuing):   c-out-8q-q1 (match-any)
  bandwidth remaining percent 15

Class-map (queuing):   c-out-8q-q-default (match-any)
  bandwidth remaining percent 30
```

Example of set qos-groups

The following is an example to set qos-groups with values 4-7.

```
switch(config)# policy-map p1
switch(config-pmap-qos)# class c1
switch(config-pmap-c-qos)# set qos-group 1
switch(config-pmap-c-qos)# ex
switch(config-pmap-qos)# class c2
switch(config-pmap-c-qos)# set qos-group 4
switch(config-pmap-c-qos)# ex
switch(config-pmap-qos)# class c3
switch(config-pmap-c-qos)# set qos-group 7
switch(config-pmap-c-qos)# ex
switch(config-pmap-qos)# ex
switch(config)# show policy-map p1


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

  policy-map type qos p1
    class  c1
      set qos-group 1
    class  c2
      set qos-group 4
    class  c3
      set qos-group 7
switch(config)# conf t
switch(config)# int ethernet 2/1
switch(config-if)# service-policy type qos input p1
switch(config-if)# show policy-map interface ethernet 2/1


Global statistics status :   enabled

Ethernet2/1

  Service-policy (qos) input:   p1
    SNMP Policy Index:  285226505

    Class-map (qos):   c1 (match-all)
      Match: dscp 10
      set qos-group 1

    Class-map (qos):   c2 (match-all)
      Match: dscp 20
      set qos-group 4
```

```
Class-map (qos):   c3 (match-all)
  Match: dscp 30
  set qos-group 7
```

Changing from 8q Mode to 4q Mode

Note Changing from 8q mode to 4q mode is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

Use the following guidelines to change from 8q mode to 4q mode:

- Ensure that none of the active input QoS policies have **set qos-group** action for qos-groups 4-7, so that no traffic flows towards queues 4-7.
- Ensure that all 8q interface policies and 8q system level policies are replaced with corresponding 4q policies.
- Replace the 8q network-qos policy with a corresponding 4q policy.

Guidelines and Limitations

Modular QoS has the following configuration guidelines and limitations:

- Eight QoS groups are supported only on modular platforms with the Cisco Nexus 9300 N9K-M4PC-CFP2 uplink module, and the following Cisco Nexus 9500 platform line cards:
  - N9K-X9432PQ
  - N9K-X9464PX
  - N9K-X9464TX
  - N9K-X9636PQ

Configuring an MQC Object

When you specify an MQC object command, the device creates the object if it does not exist and then enters map mode.

To remove a class-map or policy-map object, use the **no** form of the command that you used to create the object.

Configuring or Modifying a Class Map

You can create or modify a class map. You can then reference class maps in policy maps.

Note You cannot create a queuing class map; you must use one of the system-defined queuing class maps.

SUMMARY STEPS

1. **configure terminal**
2. **class-map type qos** [**match-any** | **match-all**] *class-name*
3. **exit**
4. **class-map type queuing match-any** *class-name*
5. **exit**
6. **show class-map** [**type qos** [*class-name*]]
7. **show class-map** [**type queuing** [*class-name*]]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map type qos** [**match-any**	**match-all**] *class-name* **Example:** `switch(config)# class-map type qos class1` `switch(config-cmap-qos)#`	Creates or accesses the class map of type qos and then enters class-map qos mode. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits class-map qos mode and enters global configuration mode.
Step 4	**class-map type queuing match-any** *class-name* **Example:** `switch(config)# class-map type queuing match-any c-out-q2` `switch(config-cmap-que)#`	Creates or accesses the class map of type queuing and then enters class-map queuing mode.
Step 5	**exit** **Example:** `switch(config-cmap-que)# exit` `switch(config)#`	Exits class map queuing mode and enters global configuration mode.
Step 6	**show class-map** [**type qos** [*class-name*]] **Example:** `switch(config)# show class-map type qos`	(Optional) Displays information about all configured class maps, all class maps of type qos, or a selected class map of type qos.
Step 7	**show class-map** [**type queuing** [*class-name*]] **Example:** `switch(config)# show class-map type queuing`	(Optional) Displays information about all configured class maps, all class maps of type queuing, or a selected class map of type queuing.

	Command or Action	Purpose
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Configuring or Modifying a Policy Map

You can create or modify a policy map that you can use to define actions to perform on class maps.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type qos** { [**match-first**] *policy-map-name*}
3. **exit**
4. **policy-map type queuing** {[**match-first**] *policy-map-name}*
5. **exit**
6. **show policy-map** [**type qos** [*policy-map-name*]]
7. **show policy-map** [**type queuing** [*policy-map-name* | *default-out-policy*]]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map type qos** { [**match-first**] *policy-map-name*} **Example:** `switch(config)# policy-map type qos policy1` `switch(config-pmap-qos)#`	Creates or accesses the policy map of type qos and then enters policy-map mode. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**exit** **Example:** `switch(config-pmap)# exit` `switch(config)#`	Exits policy-map mode and enters global configuration mode.
Step 4	**policy-map type queuing** {[**match-first**] *policy-map-name}* **Example:** `switch(config)# policy-map type queuing policy_queue1` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 5	**exit** **Example:**	Exits policy map mode and enters global configuration mode.

	Command or Action	Purpose
	`switch(config-pmap)# exit` `switch(config)#`	
Step 6	**show policy-map** [**type qos** [*policy-map-name*]] **Example:** `switch(config)# show policy-map type qos`	(Optional) Displays information about all configured policy maps, all policy maps of type qos, or a selected policy map of type qos.
Step 7	**show policy-map** [**type queuing** [*policy-map-name* | *default-out-policy*]] **Example:** `switch(config)# show policy-map type queuing`	(Optional) Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing or the default output queuing policy.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Applying Descriptions to MQC Objects

You can use the **description** command to add a description to a MQC object.

SUMMARY STEPS

1. **configure terminal**
2. Specify the MQC object whose description you want to set:
   - Class-map:

     class-map [**type qos**] [**match-any** | **match-all**] *class-name*
   - Policy-map:

     policy-map [**type qos**] [**match-first**] *policy-map-name*
3. **description** *string*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	Specify the MQC object whose description you want to set: • Class-map:	• Class-map: Creates or accesses the class map and then enters class-map mode. The class-map name can contain

	Command or Action	Purpose
	class-map [**type qos**] [**match-any** | **match-all**] *class-name* • Policy-map: **policy-map** [**type qos**] [**match-first**] *policy-map-name* **Example:** • Class-map: `switch(config-cmap)# class-map class1` `switch(config-cmap)#` • Policy-map: `switch(config)# policy-map policy1` `switch(config-pmap)#`	alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 alphanumeric characters. • Policy-map: Creates or accesses the policy map 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	**description** *string* **Example:** `switch(config-cmap)# description my traffic class` `switch(config-cmap)#`	Adds a description string to the MQC object. The description can be up to 200 alphanumeric characters. **Note** You cannot modify the description of system-defined queuing class maps.
Step 4	**exit** **Example:** `switch(config-cmap)# exit` `switch(config)#`	Exits class-map mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Verifying an MQC Object

To display MQC object configuration information, perform one of the following tasks:

Command	Purpose
show class-map [**type qos** [*class-name*]]	Displays information about all configured class maps, all class maps of type qos, or a selected class map of type qos.
show class-map [**type queuing** [*class-name*]]	Displays information about all configured class maps, all class maps of type queuing, or a selected class map of type queuing.
show policy-map [**type qos** [*policy-map-name*]]	Displays information about all configured policy maps, all policy maps of type qos, or a selected policy map of type qos.
show policy-map [**type queuing** [*policy-map-name* | **default-out-policy**]]	Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.

Attaching and Detaching a QoS Policy Action

The software does not allow you to enable or disable QoS features with a configuration command. To enable or disable QoS features, you must attach or detach QoS policies to or from interfaces or VLANs as described in this section.

The system-defined type queuing policy maps are attached to each interface unless you specifically attach a different policy map.

Note The device allows only one queuing policy per interface.

Policies that are defined at multiple interfaces have the following restrictions:

- A QoS policy attached to the physical port takes effect when the port is not a member of a port channel.
- A QoS policy attached to a port channel takes effect even when policies are attached to member ports.
- A QoS policy attached to a VLAN is applied to all ports in that VLAN that do not have other policies specifically applied.
- One ingress QoS policy is supported for each Layer 3 port and Layer 3 port-channel interface.
- One ingress QoS policy is supported for each VLAN.
- When a VLAN or port channel, or both, touches multiple forwarding engines, all policies that enforce a rate are enforced per forwarding engine.

  For example, if you configure a policer on a specific VLAN that limits the rate for the VLAN to 100 Mbps and if you configure one switch port in the VLAN on one module and another switch port in the VLAN on another module, each forwarding engine can enforce the 100-Mbps rate. In this case, you could actually have up to 200 Mbps in the VLAN that you configured to limit the rate to 100 Mbps.

Note Default queuing policies are active, unless you configure and apply another policy.

The interface where a QoS policy is applied is summarized in the following table. Each row represents the interface levels. The entry descriptions are as follows:

- Applied—Interface where an attached policy is applied.
- Present—Interface where a policy is attached but not applied.
- Not present—Interface where no policy is attached.
- Present or not—Interface where a policy is either attached or not, but not applied.

Table 13: QoS Policy Interfaces

Port Policy	Port-Channel Policy	VLAN Policy
Applied	Not present	Present or not

Present or not	Applied	Present or not
Not present	Not present	Applied

To attach a policy map to an interface or VLAN, use the **service-policy** command. The policies defined in the policy map are applied to the input stream of packets on the interface.

To detach a policy map from an interface, use the **no** form of the **service-policy** command.

Configuring a Service Policy for a Layer 2 Interface

Before you begin

Ensure that the ternary content addressable memory (TCAM) is carved for port QoS.

For more details, see the Configuring QoS TCAM Carving section.

SUMMARY STEPS

1. **configure terminal**
2. **interface interface** *slot/port*
3. **switchport**
4. **service-policy type** {**qos input** | **queuing output**} | {**qos output** | **queuing output**} *policy-map-name* [**no-stats**]
5. **show policy-map interface** *interface slot/port* **type** {**qos** | **queuing**}
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**interface interface** *slot/port* **Example:** `switch(config)# interface ethernet 1/1` `switch(config-if)#`	Enters configuration interface mode.
Step 3	**switchport** **Example:** `switch(config-if)# switchport`	Selects the Layer 2 interface.
Step 4	**service-policy type** {**qos input**	**queuing output**}	{**qos output**	**queuing output**} *policy-map-name* [**no-stats**] **Example:**	Specifies the policy map to use as the service policy for the Layer 2 interface. There are two policy-map configuration modes:

	Command or Action	Purpose
	`switch(config-if)# service-policy input policy1` `switch(config-if)#` **Example:** `switch(config-if)# interface intf1` `switch(config-if)# service-policy type qos output egressqos` `switch(config-if)# exit` `switch(config)#`	• qos input or qos output — qos input is the default classification mode. To set the classification mode to egress, use qos output. • queuing output —Queuing mode. **Note** The **output** keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply **output** to a queuing policy.
Step 5	**show policy-map interface** *interface slot/port* **type** {**qos** | **queuing**} **Example:** `switch(config)# show policy-map interface ethernet 1/1 type qos`	(Optional) Displays information about policy maps that are applied to the specified interface. You can limit what the device displays to qos or queuing policies.
Step 6	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Configuring a Service Policy for a Layer 3 Interface

Before you begin

Ensure that the ternary content addressable memory (TCAM) is carved for Layer 3 QoS.

For more details, see the Configuring QoS TCAM Carving section.

SUMMARY STEPS

1. **configure terminal**
2. **interface interface** *slot/port*
3. **no switchport**
4. **service-policy type** {**qos input** | **queuing output**} | {**qos output** | **queuing output**} *policy-map-name* [**no-stats**]
5. **show policy-map interface** *interface slot/port* **type** {**qos** | **queuing**}
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface interface** *slot/port* **Example:** `switch(config)# interface ethernet 1/1` `switch(config-if)#`	Enters configuration interface mode.
Step 3	**no switchport** **Example:** `switch(config-if)# no switchport`	Selects the Layer 3 interface.
Step 4	**service-policy type** {**qos input** | **queuing output**} | {**qos output** | **queuing output**} *policy-map-name* [**no-stats**] **Example:** `switch(config-if)# service-policy input policy1` `switch(config-if)#` **Example:** `switch(config-if)# service-policy output policy1` `switch(config-if)#`	Specifies the policy map to use as the service policy for the Layer 3 interface. There are two policy-map configuration modes: • qos input or qos output — qos input is the default classification mode. To set the classification mode to egress, use qos output. • queuing output —Queuing mode. **Note** The **output** keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply **output** to a queuing policy.
Step 5	**show policy-map interface** *interface slot/port* **type** {**qos** | **queuing**} **Example:** `switch(config)# show policy-map interface ethernet 1/1 type qos`	(Optional) Displays information about policy maps that are applied to the specified interface. You can limit what the device displays to qos or queuing policies.
Step 6	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

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. **configure terminal**
2. **system qos**
3. **service-policy type** {**network-qos** | **queuing output**} *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**system qos** **Example:** `switch(config)# system qos` `switch(config-sys-qos)#`	Enters system class configuration mode.
Step 3	**service-policy type** {**network-qos** | **queuing output**} *policy-map-name* **Example:** `switch(config-sys-qos)# service-policy input default-nq-policy`	Specifies the policy map to use as the service policy (default-nq-policy) for the system. There are two policy-map configuration modes: • network-qos—Network-wide (system qos) mode. **Note** To restore the system to the default service policies, use the **no** form of the command. • queuing—Queuing mode (output at system qos and interface). **Note** There is no default policy-map configuration mode. You must specify the type. The **output** keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply **output** to a queuing policy.

Attaching a QoS Policy Action to a VLAN

Before you begin

Ensure that the ternary content-addressable memory (TCAM) is carved for VLAN QoS.

For more details, see the QoS TCAM carving chapter.

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan-id-list*
3. **service-policy** [**type qos**] {**input**} | {**qos output** } {*policy-map-name*} [**no-stats**]
4. **show policy-map** [**interface** *interface* | **vlan** *vlan-id*] [**input**] [**type qos** | **queuing**] [**class** [**type qos** | **queuing**] *class-map-name*]
5. **copy running-config startup-config**

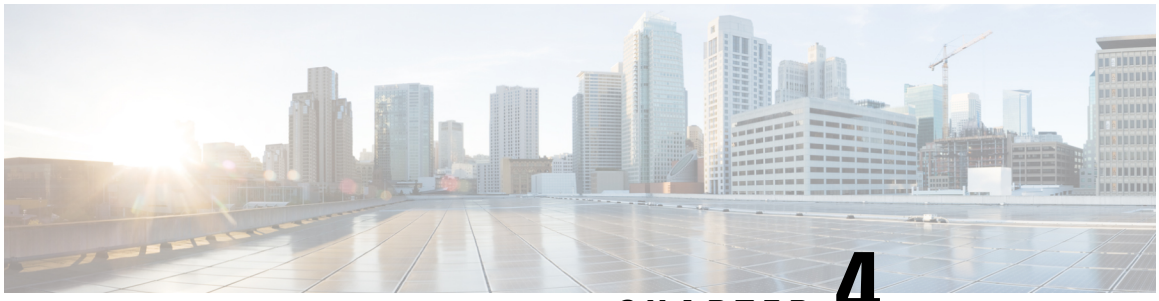
DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**vlan configuration** *vlan-id-list* **Example:** `switch(config)# vlan configuration 2` `switch(config-vlan-config)#`	Enters VLAN configuration mode. **Note** *vlan-id-list* is a space-separated list of VLANs.
Step 3	**service-policy** [**type qos**] {**input**} | {**qos output** } {*policy-map-name*} [**no-stats**] **Example:** `switch(config-vlan-config)# service-policy type qos input policy1` **Example:** `switch(config-if)# service-policy type qos output egressqos` `switch(config-if)# exit` `switch(config)#`	Adds the policy map to the input packets of a VLAN. Only one input policy can be attached to a VLAN. The example adds policy1 to the VLAN. Label sharing only occurs when QoS policies under VLANs are configured with the **no-stats** option. With the **no-stats** option, the QoS label gets shared when the same QoS policy is applied on multiple VLANs. **Note** When the **no-stats** option is configured, the ingress QoS policy-map statistics on a VLAN basis are not available because the label is shared.
Step 4	**show policy-map** [**interface** *interface* | **vlan** *vlan-id*] [**input**] [**type qos** | **queuing**] [**class** [**type qos** | **queuing**] *class-map-name*] **Example:** `switch(config)# show policy-map vlan 2`	(Optional) Displays information about policy maps that are applied to all interfaces or the specified interface. You can limit what the device displays to input policies, qos or queuing polices, and to a specific class.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Session Manager Support for QoS

Session Manger supports the configuration of QoS. This feature allows you to verify the QoS configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. For information about Session Manager, see the *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*.

After you start the configuration session, do not enter any configuration commands using the configure terminal configuration mode until the configuration session is aborted or committed. Entering parallel configurations (one configuration that uses the configuration session and another using the configuration terminal configuration mode) might cause verification failures in the configuration session mode.



CHAPTER 4

Configuring QoS TCAM Carving

About QoS TCAM Carving

You can change the size of the access control list (ACL) ternary content addressable memory (TCAM) regions in the hardware.

On Cisco Nexus 9300 and 9500 platform switches and Cisco Nexus 3164Q, 31128PQ, 3232C, and 3264Q switches, the egress TCAM size is 1K, divided into four 256 entries. On other Cisco Nexus 9300 and 9500 platform switches and Cisco Nexus 3164Q and 31128PQ switches, the ingress TCAM size is 4K, divided into eight 256 slices and four 512 slices. A slice is the unit of allocation. A slice can be allocated to one region only. For example, a 512-size slice cannot be used to configure two features of size 256 each. Similarly, a 256-size slice cannot be used to configure two features of size 128 each. The IPv4 TCAM regions are single wide. The IPv6, QoS, MAC, CoPP, and system TCAM regions are double wide and consume double the physical TCAM entries. For example, a logical region size of 256 entries actually consumes 512 physical TCAM entries.

The number of default entries for QoS TCAM carving are:

- The default QoS TCAM carving for the Cisco Nexus 9504, Cisco Nexus 9508, and Cisco Nexus 9516 is for Layer 3 QoS (IPV4) with 256 entries. For these switches, all of the QoS TCAM entries are double wide.
- The default QoS TCAM carving for ALE (Application Leaf Engine) enabled devices is for Layer 2 port QoS (IPV4) with 256 entries. For these switches, all of the QoS TCAM entries are double wide.

Note In addition to the above TCAM, for ALE enabled devices, a separate TCAM in the Cisco Nexus C9396PX (uplink ports) and Cisco Nexus C93128TX (uplink ports) ASIC is used for the QoS classification policies applied on 40G uplink ports. By default, this separate TCAM is carved for Layer 3 QoS (IPV4), Layer 2 Port QoS (IPV4), and VLAN QoS (IPV4) with 256 entries each.

Table 14: QoS TCAM Regions (CIsco NX-OS Release 7.1(3)I6(1))

Feature	Purpose	Region Name
Egress QoS	QoS policy applied on interfaces in output direction.	IPV4: e-qos Cisco Nexus 922 series switch: egr-l2-qos, egr-l3-vlan-qos IPV6: e-ipv6-qos MAC: e-mac-qos See notes following table.

Table 15: QoS TCAM Regions (Cisco NX-OS Release 6.1(2)I3(4) and earlier)

Feature	Purpose	Region Name
Layer 3 QoS	QoS policy applied on Layer 3 interfaces.	IPV4: l3qos*, ns-l3qos* IPV6: ipv6-l3qos*, ns-ipv6-l3qos* See notes following table.
Port QoS	QoS policy applied on Layer 2 interfaces.	IPV4: qos*, ns-qos* IPV6: ipv6-qos*, ns-ipv6-qos* MAC: mac-qos*, ns-mac-qos* See notes following table.
VLAN QoS	QoS policy applied on VLAN.	IPV4: vqos, ns-vqos IPV6: ipv6-vqos*, ns-ipv6-vqos* MAC: mac-vqos*, ns-mac-vqos* See notes following table.
FEX QoS	QoS policy applied on FEX interfaces.	IPV4: fex-qos* IPv6: fex-ipv6-qos* MAC: fex-mac-qos* See notes following table.

Note * The region is applicable only for ALE enabled devices and are required for classification policies applied on 40G uplink ports.

You need to save the configuration and reload the system for the region configuration to become effective.

About QoS TCAM Lite Regions

IPV4 requires QoS TCAM regions to be double wide TCAMs to support conform/violate policer statistics. If conform/violate statistics are not required, the size of the QoS TCAM entries can be reduced to single wide TCAMs by using QoS TCAM lite regions. Policing is supported by these regions, however only violate packets/bytes statistics are supported.

Table 16: QoS TCAM Regions (Release 7.1(3)I6(1))

Feature	Purpose	Region Name
Egress QoS	QoS policy applied on interfaces in output direction.	IPV4: e-qos-lite See notes following table.

Table 17: QoS TCAM Lite Regions

Feature	Purpose	Region Name
Layer 3 QoS	QoS policy applied on Layer 3 interfaces.	IPV4: l3qos-lite
Port QoS	QoS policy applied on Layer 2 interfaces.	IPV4: qos-lite
VLAN QoS	QoS policy applied on VLAN.	IPV4: vqos-lite
FEX QoS	QoS policy applied on FEX interfaces.	IPV4: fex-qos-lite

Note The region is applicable only for ALE enabled devices and are required for classification policies applied on 40G uplink ports.

You need to save the configuration and reload the system for the region configuration to become effective.

Note Either the regular version or the lite version of the QOS TCAM can be enabled. Both cannot be enabled at the same time. For example, either the IPv4 Port QoS or the IPv4 Port QoS lite version can be enabled at any one time.

Guidelines and Limitations

TCAM region sizes have the following configuration guidelines and limitations:

- **show** commands with the **internal** keyword are not supported.
- After TCAM carving, you must save the configuration and reload the switch.

- Cisco Nexus 9200 platform switches and Cisco Nexus 9300-EX platform switches are of the same type and therefore, they have the same TCAM regions.
- By default, all IPv6 TCAMs are disabled (the TCAM size is set to 0).
- Use the **show hardware access-list tcam region** command to view the configured TCAM region size.
- By default, the TCAM region for CoPP is 95% utilized on the Nexus 9300/Nexus 9500 platform switch. If you modify the CoPP policy, it is likely that you will need to modify other TCAM region sizes to allow for more space to be applied to the CoPP TCAM region.
- When any of the following classification criteria are used for IPv4 and IPv6, you need to carve the IPv4 based QoS TCAM region. It is not necessary to carve an IPv6 based QoS TCAM region.
  - Differentiated Services Code Point (DSCP) based classification
  - Class of service (CoS) based classification
  - IP precedence based classification
- When a QoS policy is applied on multiple interfaces or multiple VLANs, the label is not shared since the statistics option is enabled.

  To share the label for the same qos policy that is applied on multiple interfaces or multiple VLANs, you need to configure the qos policy with no-stats option using the **service-policy type qos input my-policy no-stats** command.
- On Cisco Nexus 9300 platform switches, the Cisco Nexus 9536PQ, 9564PX, and 9564TX line cards are used to enforce the QoS classification policies applied on 40G ports. It has 768 TCAM entries available for carving in 256-entry granularity. These region names are prefixed with "ns-".
- For the Cisco Nexus 9536PQ, 9564PX, and 9564TX line cards, only the IPv6 TCAM regions consume double-wide entries. The rest of the TCAM regions consume single-wide entries.
- When a VACL region is configured, it is configured with the same size in both the ingress and egress directions. If the region size cannot fit in either direction, the configuration is rejected.
- On the Cisco Nexus 9508 switch with the -R series line card, VLAN QoS is only supported with Cisco NX-OS Release 7.0(3)F3(3) and later releases.

Configuring QoS TCAM Carving

You can change the default QoS TCAM carving to accommodate your network requirements. The following sections contain examples of how to change the default QoS TCAM carving.

Enabling Layer 3 QoS (IPv6)

The default TCAM region configuration does not accommodate Layer 3 QoS (IPv6). To enable Layer 3 QoS (IPv6), you must decrease the TCAM size of another region and then increase the TCAM size to enable the new Layer 3 QoS (IPv6) region.

Table 18: Default TCAM Region Configuration (Ingress) for the Cisco Nexus 9504, Cisco Nexus 9508, and Cisco Nexus 9516 devices

Region Name	Size	Width	Total Size
IPV4 RACL	1536	1	1536
L3 QoS(IPV4)	256	2	512
COPP	256	2	512
System	256	2	512
Redirect	256	1	256
SPAN	256	1	256
VPC Convergence	512	1	512
			4K

Procedure

	Command or Action	Purpose
Step 1	**hardware access-list tcam region** *region tcam-size*	To enable carving your Layer 3 QoS (IPv6) TCAM region, specify another region to free up resources. Also specify the reduced TCAM size for the region. **Note** Repeat this step for as many regions as necessary to free up sufficient resources to carve the new Layer 3 QoS (IPv6) TCAM region.
Step 2	**hardware access-list tcam region** *region tcam-size*	Carve the new Layer 3 QoS (IPv6) TCAM region including the TCAM size (number of double wide entries).

Example

This example sets the ingress Layer 3 QoS (IPv6) TCAM region size to 256. A Layer 3 QoS (IPv6) of size 256 takes 512 entries because IPv6 is double wide.

- Reduce the span and redirect regions to 0. This creates 512 entry spaces that are used to carve Layer 3 QoS (IPV6) with 256 entries (double wide).

```
switch(config)# hardware access-list tcam region redirect 0
 Warning: Please reload the linecard for the configuration to take effect
 Warning: BFD, DHCPv4 and DHCPv6 features will NOT be supported after this configuration
change.
switch(config)# hardware access-list tcam region span 0
 Warning: Please reload the linecard for the configuration to take effect
switch(config)# hardware access-list tcam region ipv6-l3qos 256
 Warning: Please reload the linecard for the configuration to take effect
```

Table 19: Updated TCAM Region Configuration After Reducing the IPv4 RACL (Ingress)

Region Name	Size	Width	Total Size
IPv4 RACL	1536	1	1536
Layer 3 QoS (IPv6)	256	2	512
Layer 3 QoS (IPv4)	256	2	512
CoPP	256	2	512
System	256	2	512
Redirect	0	1	0
SPAN	0	1	0
VPC Convergence	512	1	512
			4K

Enabling VLAN QoS (IPv4)

To enable VLAN QoS (IPv4), you must decrease the TCAM size of another region and then increase the TCAM size to enable the new VLAN QoS (IPv4) region.

The following table list the default sizes for the ingress TCAM regions for ALE enabled devices.

Table 20: Default TCAM Region Configuration (Ingress)

Region Name	Size	Width	Total Size
PACL (IPV4)	512	1	512
Port QoS (IPV4)	256	2	512
VACL (IPV4)	512	1	512
RACL(IPV4)	512	1	512
System	256	2	512
COPP	256	2	512
Redirect	512	1	512
SPAN	256	1	256
VPC Converg	256	1	256
			4K

Procedure

	Command or Action	Purpose
Step 1	**hardware access-list tcam region** *region tcam-size*	To enable carving for your VLAN QoS (IPv4) TCAM region, specify another region to free up resources. Also specify the reduced TCAM size for the region. **Note** Repeat this step for as many regions as necessary to free up sufficient resources to carve the new VLAN QoS (IPv4) TCAM region.
Step 2	**hardware access-list tcam region** *region tcam-size*	Carve the new VLAN QoS (IPv4) TCAM region including the TCAM size (number of double wide entries).

Example

This example sets the VLAN QoS (IPv4) TCAM size to 256. A VLAN QoS (IPv4) of size 256 takes 512 entries because QoS TCAM is double wide.

- Reduce the ingress Port QoS (IPv4) by 256 bytes (QoS features are double wide, 2 x 256 = 512) and add an ingress VLAN QoS (IPv4) with 256 (2 x 256).

```
switch(config)# hardware access-list tcam region qos 0
 Warning: Please reload the linecard for the configuration to take effect
switch(config)# hardware access-list tcam region vqos 256
 Warning: Please reload the linecard for the configuration to take effect
```

Table 21: Updated TCAM Region Configuration After Reducing the IPv4 Port QoS Ingress

Region Name	Size	Width	Total Size
PACL (IPV4)	512	1	512
Port QoS (IPV4)	0	2	0
VLAN QoS(IPV4)	256	2	512
VACL (IPV4)	512	1	512
RACL(IPV4)	512	1	512
System	256	2	512
COPP	256	2	512
Redirect	512	1	512
SPAN	256	1	256
VPC Converg	256	1	256
			4K

Enabling FEX QoS (IPv4)

Note The FEX QoS feature is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

To enable FEX QoS (IPv4), you must decrease the TCAM size of another region and then increase the TCAM size to enable the new FEX QoS (IPv4) region.

Procedure

	Command or Action	Purpose
Step 1	**hardware access-list tcam region** *region tcam-size*	To enable carving your FEX QoS (IPv4) TCAM region, specify another region to free up resources. Also specify the reduced TCAM size for the region. **Note** Repeat this step for as many regions as necessary to free up sufficient resources to carve the new FEX QoS (IPv4) TCAM region.
Step 2	**hardware access-list tcam region** *region tcam-size*	Carve the new FEX QoS (IPv4) TCAM region including the TCAM size (number of double wide entries).

Example

This example sets the FEX QoS (IPv4) TCAM size to 256. A FEX QoS (IPv4) of size 256 takes 512 entries because QoS TCAM is double wide.

- Reduce the IPv4 FEX IFACL region by 512 entries and add a FEX QoS (IPv4) region with 512 entries.

```
switch(config)# hardware access-list tcam region fex-ifacl 0
 Warning: Please reload the linecard for the configuration to take effect
switch(config)# hardware access-list tcam region fex-qos 256
 Warning: Please reload the linecard for the configuration to take effect
```

Verifying QoS TCAM Carving

After you adjust the TCAM region sizes, enter the **show hardware access-list tcam region** command to display the TCAM sizes that will be applicable on the next reload of the device.

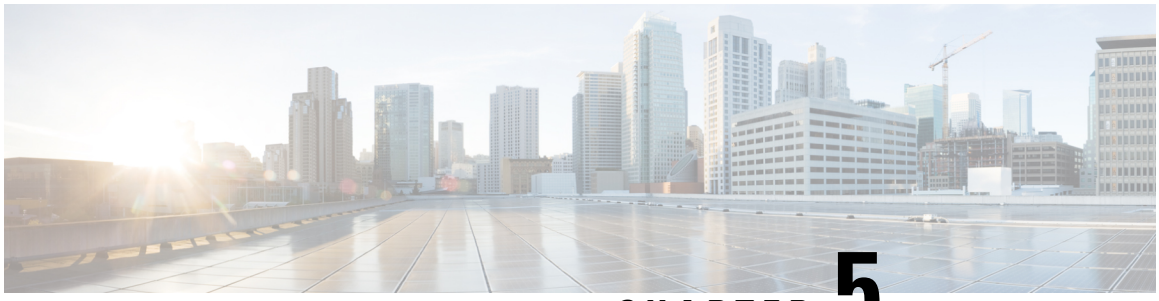
Note To keep all modules synchronized, you must reload all line card modules or enter the **copy running-config startup-config** command and the **reload** command to reload the device. Multiple TCAM region configurations require only a single reload. You can wait until you complete all of your TCAM region configurations before you reload the device.

If you exceed the 4K ingress limit for all TCAM regions when you configure a TCAM region, the following message appears:

```
ERROR: Aggregate TCAM region configuration exceeded the available Ingress TCAM space.
Please re-configure.
```

If TCAM for a particular feature is not configured and you try to apply a feature that requires TCAM carving, the following message appears:

```
ERROR: Module x returned status: TCAM region is not configured. Please configure TCAM
region and retry the command.
```



CHAPTER 5

Configuring Classification

About Classification

Classification is the separation of packets into traffic classes. You configure the device to take a specific action on the specified classified traffic, such as policing or marking down, or other actions.

You can create class maps to represent each traffic class by matching packet characteristics with the classification criteria in the following table:

Table 22: Classification Criteria

Classification Criteria	Description
CoS	Class of service (CoS) field in the IEEE 802.1Q header.
IP precedence	Precedence value within the type of service (ToS) byte of the IP header.
Differentiated Services Code Point (DSCP)	DSCP value within the DiffServ field of the IP header.
ACL	IP, IPv6, or MAC ACL name.
Packet length	Size range of Layer 3 packet lengths.
IP RTP	Identify applications using Real-time Transport Protocol (RTP) by UDP port number range.

You can specify multiple match criteria, you can choose to not match on a particular criterion, or you can determine the traffic class by matching any or all criteria.

Note However, if you match on an ACL, no other match criteria, except the packet length, can be specified in a match-all class. In a match-any class, you can match on ACLs and any other match criteria.

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

You can reuse class maps when defining the QoS policies for different interfaces that process the same types of traffic.

Prerequisites for Classification

Classification has the following prerequisites:

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

Guidelines and Limitations

Classification has the following configuration guidelines and limitations:

- The **show** commands with the **internal** keyword are not supported.
- When the **destination interface sup-eth0** CLI command is configured, the following system log message is displayed: `Enabling span destination to SUP will affect ingress QoS classification.`
- For VXLAN, beginning Cisco NX-OS Release 7.0(3)I6(1), the following Cisco Nexus switches support QoS policies for traffic in the network to host direction (decapsulation path) as egress policy on both the port and VLAN:
  - Cisco Nexus 9300 and 9500 platform switches.
  - Cisco Nexus 9200 and 9300-EX platform switches; Cisco Nexus 93180YC-EX and 93108TC-EX switches; and the Cisco Nexus 9732C-EX line card.
  - The above is not supported for the following hardware: Cisco Nexus 9230QC, 9272Q, 9232C, 9236C, and 92300YC switches; and Cisco Nexus 9160YC-X switches.
- For VXLAN, beginning Cisco NX-OS Release 7.0(3)I6(1), the following Cisco Nexus switches do not support QoS policies for traffic from the network to access direction (decapsulation path) as ingress policy on the uplink interface:
  - Cisco Nexus 9300 and 9500 platform switches.
  - Cisco Nexus 9200 and 9300-EX platform switches; and Cisco Nexus 93180YC-EX and 93108TC-EX switches; and the Cisco Nexus 9732C-EX line card.
  - Cisco Nexus 9230QC, 9272Q, 9232C, 9236C, and 92300YC switches; and Cisco Nexus 9160YC-X switches.

- For matching the packets based on DSCP, CoS, or precedence in Cisco Nexus 9300-EX platform switches, the TCAM entries for both IPv4 (single-wide is 1 entry) and IPv6 (double-wide are 2 entries) are installed in the hardware. For example, if you match DSCP 4, 3 entries are installed in the hardware, 1 entry for IPv4 and 2 entries for IPv6.
- You can specify a maximum of 1024 match criteria in a class map.
- You can configure a maximum of 128 classes for use in a single policy map.
- When you match on an ACL, the only other match you can specify is the Layer 3 packet length in a match-all class.
- The **match-all** option in the **class-map type qos match-all** command is not supported. The match criteria of this command becomes the same as in the **class-map type qos match-any** command. The **class-map type qos match-all** command yields the same results as the **class-map type qos match-any** command.
- You can classify traffic on Layer 2 ports based on either the port policy or VLAN policy of the incoming packet but not both. If both are present, the device acts on the port policy and ignores the VLAN policy.
- When a Cisco Nexus Fabric Extender (FEX) is connected and in use, data traffic should not be marked with a CoS value of 7. CoS 7 is reserved for control traffic transiting the Fabric Extender.
- Control traffic (control frames) from the switch to the FEX are marked with a CoS value of 7 and are limited to a jumbo MTU frame size of 2344 bytes.
- FEX host interfaces (HIF) are supported by the FEX QoS policy.
  - QoS TCAM carving is supported on ALE (Application Leaf Engine) enabled switches.
  - Only system level policies are supported.
  - Match on CoS is supported.
  - Match on QoS-group is supported.
- A jumbo ping (MTU of 2400 or greater) from a switch supervisor with a COS of 7 to a FEX host fails because the control queue on a FEX supports an MTU limited to 2240.
- QoS classification policies are not supported under system qos for Layer 2 switch ports. However, you can configure a QoS policy to classify the incoming traffic based on CoS/DSCP and map it to different queues. The QoS policy needs to be applied under all the interfaces that require the classification.
- As a best practice, avoid having a voice VLAN configuration where an access VLAN is same as the voice VLAN.

  The following are alternative approaches:

  - If a separate dot1p tag (cos) value is not required for voice traffic, use the **switchport voice vlan untagged** command.

    ```
    switch(config)# interface ethernet 1/1
    switch(config-if)# switchport access vlan 20
    switch(config-if)# switchport voice vlan untagged
    ```

  - If a separate cos value is required for voice traffic, use the **switchport voice vlan dot1p** command.

    ```
    switch(config)# interface ethernet 1/1
    switch(config-if)# switchport access vlan 20
    switch(config-if)# switchport voice vlan dot1p
    ```

- Cisco Nexus 9504 and Cisco Nexus 9508 switches with the following line cards do not support QoS match acl with fragments:
  - Cisco Nexus 96136YC-R
  - Cisco Nexus 9636C-RX
  - Cisco Nexus 9636Q-R
  - Cisco Nexus 9636C-R
- Ingress DROP_ACL_DROP is seen with Cisco Nexus 9272Q, 9236C, and 92160YC-X switches on an ASIC during congestion. However, these drops do not impact the performance of the switch.
- MPLS packets with a NULL label on transit nodes, receive an MPLS classification based on its NULL label EXP.
- Ingress DROP_ACL_DROP is seen with Cisco Nexus 9272Q, 9236C, and 92160YC-X switches on an ASIC during congestion. However, these drops do not impact the performance of the switch.

Configuring Traffic Classes

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 though a match criteria in the access-list has a deny action, it is still used for matching for this class.

Note Use the **class-map class_acl** command to display the ACL class-map configuration.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match access-group name** *acl-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any**	**match-all**] *class-name* **Example:** `switch(config)# class-map class_acl`	Creates or accesses the class map named class-name and enters class-map mode. The class map name can contain alphabetic, hyphen, or underscore characters, and can be

	Command or Action	Purpose
		up to 40 characters. (**match-any** is the default when no option is selected and multiple match statements are entered.)
Step 3	**match access-group name** *acl-name* **Example:** `switch(config-cmap-qos)# match access-group name my_acl`	Configures the traffic class by matching packets based on the *acl-name*. The **permit** and **deny** ACL keywords are ignored in the matching.

Examples: Configuring ACL Classification

To prevent packets from being matched by the QoS class-map, you must explicitly specify the packets you want to match with permit statements. The *implicit* default deny statement at the end of the ACL will filter out the remainder. Any *explicit* deny statements configured inside the access list of a QoS class map will be ignored in the matching and treated as an explicit permit statement as shown in the examples below.

The following examples, A1, B1, and C1, all produce the same QoS matching results:

- A1

```
ip access-list extended A1
  permit ip 10.1.0.0 0.0.255.255 any
  permit ip 172.16.128.0 0.0.1.255 any
  permit ip 192.168.17.0 0.0.0.255 any
```

- B1

```
ip access-list extended B1
  permit ip 10.1.0.0 0.0.255.255 any
  deny ip 172.16.128.0 0.0.1.255 any  /* deny is interpreted as a permit */
  permit ip 192.168.17.0 0.0.0.255 any
```

- C1

```
ip access-list extended C1
   deny ip 10.1.0.0 0.0.255.255 any   /* deny is interpreted as a permit */
 deny ip 172.16.128.0 0.0.1.255 any  /* deny is interpreted as a permit */
 deny ip 192.168.17.0 0.0.0.255 any  /* deny is interpreted as a permit */
```

Adding an explicit DENY ALL at the end of a QoS matching ACL causes the QoS ACL to permit all traffic.

The following examples, D1 and E1, produce the same QoS matching results:

- D1

```
ip access-list extended D1
  permit ip 10.1.0.0 0.0.255.255 any
  permit ip 172.16.128.0 0.0.1.255 any
  permit ip 192.168.17.0 0.0.0.255 any
 deny ip 0.0.0.0 255.255.255.255 any  /* deny is interpreted as a permit */
```

Note The last line in the example effectively becomes a PERMIT ALL statement and results in the QoS ACL to permit all packets.

- E1

```
ip access-list extended E1
    permit ip 0.0.0.0 255.255.255.255 any
```

Configuring DSCP Classification

You can classify traffic based on the DSCP value in the DiffServ field of the IP header. The standard DSCP values are listed in the following table:

Table 23: Standard DSCP Values

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

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

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match** [**not**] **dscp** *dscp-values*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any**	**match-all**] *class-name* **Example:** `switch(config)# class-map class_dscp`	Creates or accesses the class map named class-name and enters class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, and can be up to 40 characters.
Step 3	**match** [**not**] **dscp** *dscp-values* **Example:** `switch(config-cmap-qos)# match dscp af21, af32`	Configures the traffic class by matching packets based on dscp-values. The standard DSCP values are shown in the following table. Use the **not** keyword to match on values that do not match the specified range.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits global class-map queuing mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to display the DSCP class-map configuration:

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

Configuring IP Precedence Classification

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

Table 24: Precedence Values

Value	List of Precedence Values
0-7	IP precedence value
critical	Critical precedence (5)
flash	Flash precedence (3)
flash-override	Flash override precedence (4)
immediate	Immediate precedence (2)
internet	Internetwork control precedence (6)
network	Network control precedence (7)
priority	Priority precedence (1)
routine	Routine precedence (0)

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match** [**not**] **precedence** *precedence-values*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any** | **match-all**] *class-name* **Example:** `switch(config)# class-map class_ip_precedence`	Creates or accesses the class map named class-name and then enters class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, and can be up to 40 characters.

	Command or Action	Purpose
Step 3	**match** [**not**] **precedence** *precedence-values* **Example:** `switch(config-cmap-qos)# match precedence 1-2, 5-7`	Configures the traffic class by matching packets based on *precedence-values*. Values are shown in the following table. Use the **not** keyword to match on values that do not match the specified range.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits global class-map queuing mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to display the IP precedence class-map configuration:

```
switch# show class-map class_ip_precedence
```

Configuring Protocol Classification

For Layer 3 protocol traffic, you can use the ACL classification match.

Table 25: match Command Protocol Arguments

Argument	Description
arp	Address Resolution Protocol (ARP)
bridging	Bridging
cdp	Cisco Discovery Protocol (CDP)
dhcp	Dynamic Host Configuration (DHCP)
isis	Intermediate system to intermediate system (IS-IS)

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match** [**not**] **protocol** {**arp** | **bridging** | **cdp** | **dhcp** | **isis**}
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any** | **match-all**] *class-name* **Example:** `switch(config)# class-map class_protocol`	Creates or accesses the class map named class-name and then enters class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, and can be up to 40 characters.
Step 3	**match** [**not**] **protocol** {**arp** | **bridging** | **cdp** | **dhcp** | **isis**} **Example:** `switch(config-cmap-qos)# match protocol isis`	Configures the traffic class by matching packets based on the specified protocol. Use the **not** keyword to match on protocols that do not match the protocol specified.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits global class-map queuing mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to display the protocol class-map configuration:

```
switch# show class-map class_protocol
```

Configuring Layer 3 Packet Length Classification

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

Note This feature is designed for IP packets only.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match** [**not**] **packet length** *packet-length-list*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any** | **match-all**] *class-name* **Example:** `switch(config)# class-map class_packet_length`	Creates or accesses the class map named class-name and then enters class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, and can be up to 40 characters.
Step 3	**match** [**not**] **packet length** *packet-length-list* **Example:** `switch(config-cmap-qos)# match packet length min 2000`	Configures the traffic class by matching packets based on various packet lengths (bytes). Values can range from 1 to 9198. Use the **not** keyword to match on values that do not match the specified range.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits global class-map queuing mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to display the packet length class-map configuration:

```
switch# show class-map class_packet_length
```

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. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match** [**not**] **cos** *cos-list*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any** | **match-all**] *class-name* **Example:** `switch(config)# class-map class_cos`	Creates or accesses the class map named class-name and then enters class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, and can be up to 40 characters.
Step 3	**match** [**not**] **cos** *cos-list* **Example:** `switch(config-cmap-qos)# match cos 4,5-6`	Configures the traffic class by matching packets based on the list of CoS values. Values can range from 0 to 7. Use the **not** keyword to match on values that do not match the specified range. **Note** When a Cisco Nexus Fabric Extender (FEX) is connected and in use, data traffic should not be marked with a CoS value of 7. CoS 7 is reserved for control traffic transiting the Fabric Extender.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits global class-map queuing mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to display the CoS class-map configuration:

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

Configuring CoS Classification for FEX

Note The CoS Classification for FEX feature is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

You can classify traffic based on the class of service (CoS) for a FEX.

Before you begin

Before configuring the FEX, enable **feature-set fex**.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match** [**not**] **cos** *cos-list*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any** | **match-all**] *class-name* **Example:** `switch(config)# class-map class_cos`	Creates or accesses the class map named class-name and then enters class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, and can be up to 40 characters.
Step 3	**match** [**not**] **cos** *cos-list* **Example:** `switch(config-cmap-qos)# match cos 4,5-6`	Configures the traffic class by matching packets based on the list of CoS values. Values can range from 0 to 7. Use the **not** keyword to match on values that do not match the specified range. **Note** When a Cisco Nexus Fabric Extender (FEX) is connected and in use, data traffic should not be marked with a CoS value of 7. CoS 7 is reserved for control traffic transiting the Fabric Extender.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits global class-map queuing mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to configure the CoS class-map configuration:

```
switch# conf t
switch(config)# class-map type qos match-all cos6
switch(config-cmap-qos)# match cos 6
switch(config)# class-map type qos match-all cos1
switch(config-cmap-qos)# match cos 1
switch(config)# class-map type qos match-all cos2
switch(config-cmap-qos)# match cos 2
switch(config)# class-map type qos match-all cos3
switch(config-cmap-qos)# match cos 3
switch(config)# class-map type qos match-all cos0
switch(config-cmap-qos)# match cos 0
```

Configuring IP RTP Classification

The IP Real-Time Transport Protocol (RTP) is a transport protocol for real-time applications that transmit data such as audio or video (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 uses an even-numbered port and the next higher odd-numbered port is used for RTP Control Protocol (RTCP) communications.

When defining a match statement in a **type qos class-map**, to match with upper layer protocols and port ranges (UDP/TCP/RTP, among others), the system cannot differentiate, for example, between UDP traffic and RTP traffic in the same port range. The system classifies both traffic types the same. For better results, you must engineer the QoS configurations to match the traffic types present in the environment.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**type qos**] [**match-any** | **match-all**] *class-name*
3. **match** [**not**] **ip rtp** *udp-port-value*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map** [**type qos**] [**match-any**	**match-all**] *class-name* **Example:** `switch(config)# class-map class_rtp`	Creates or accesses a class map and then enters the class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, and can be up to 40 characters.
Step 3	**match** [**not**] **ip rtp** *udp-port-value* **Example:** `switch(config-cmap-qos)# match ip rtp 2000-2100, 4000-4100`	Configures the traffic class by matching packets that are based on a range of lower and upper UDP port numbers, targeting applications using RTP. Values can range from 2000 to 65535. Use the **not** keyword to match on values that do not match the specified range.

	Command or Action	Purpose
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits global class-map queuing mode and enters global configuration mode.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to display the RTP class-map configuration:

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

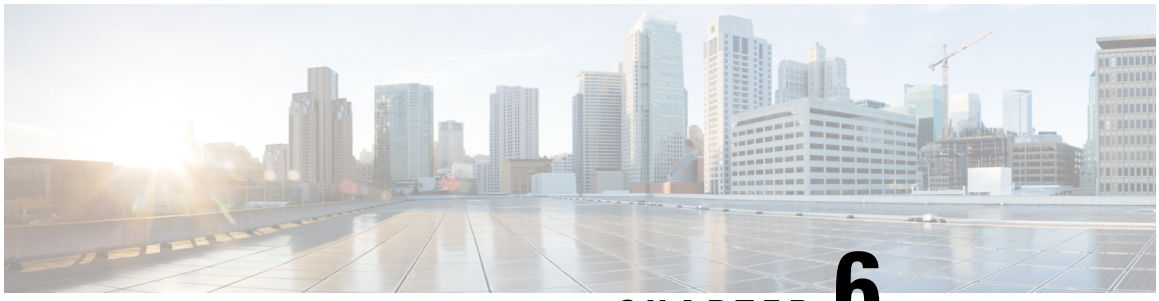
Verifying the Classification Configuration

Use the **show class-map** command to verify the class-map configuration. This command displays all class maps.

Configuration Examples for Classification

The following example shows how to configure classification for two classes of traffic:

```
class-map class_dscp
match dscp af21, af32
exit
class-map class_cos
match cos 4, 5-6
exit
```



CHAPTER 6

Configuring Marking

About Marking

Marking is a method that you use to modify the QoS fields of the incoming and outgoing packets. The QoS fields that you can mark are IP precedence and differentiated services code point (DSCP) in Layer 3. The QoS group is a label local to the system to which you can assign intermediate marking values. You can use the QoS group label to determine the egress scheduling.

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

Table 26: Configurable Marking Features

Marking Feature	Description
DSCP	Layer 3 DSCP.
IP precedence	Layer 3 IP precedence. **Note** IP precedence uses only the lower three bits of the type of service (ToS) field. The device overwrites the first three bits of the ToS field to 0.
QoS group	Locally significant QoS values that can be manipulated and matched within the system. The range is from 0 to 3.
Ingress	Status of the marking applies to incoming packets.
CoS	Layer 2 VLAN ID

Trust Boundaries

The trust boundary forms a perimeter on your network. Your network trusts (and does not override) the markings on your switch.

The incoming interface enforces the trust boundary as follows:

- All Fibre Channel and virtual Fibre Channel interfaces are automatically classified into the FCoE system class.
- By default, all Ethernet interfaces are trusted interfaces. A packet tagged with an 802.1p class of service (CoS) value is classified into a system class using the value in the packet.
- Any packet not tagged with an 802.1p CoS value is classified into the default drop system class. If the untagged packet is sent over a trunk, it is tagged with the default untagged CoS value, which is zero.
- You can override the default untagged CoS value for an Ethernet interface or port channel.

After the system applies the correct CoS value to an untagged packet, QoS treats the packet according to the newly defined class.

Class of Behavior

For routed unicast traffic, the CoS value is not available and the packet has the Differentiated Services Code Point (DSCP) value only. For bridged unicast traffic, the CoS value is copied from the CoS value received in the 802.1q header. Note that on Layer 2 access links there is no trunk header. Therefore, if traffic is received on an access port and bridged, it will egress the switch with CoS 0. The DSCP value does not change, but the packet may not get the desired priority. You can manually set the CoS value in a policy-map via any QoS policy that manually sets the CoS or DSCP value.

Routed multicast traffic derives its CoS value similar to routed unicast traffic. For bridged multicast traffic, the behavior depends on the Layer 3 state. If there is no Layer 3 state for the multicast group, the CoS is derived similar to the bridged unicast traffic. If there is a Layer 3 state for the multicast group, the CoS is derived similar to routed unicast traffic.

Note When you enable Protocol Independent Multicast (PIM) in sparse mode on the switch virtual interface (SVI) for the VLAN in which traffic is received, PIM creates an S,G entry for any multicast traffic.

Table 27: CoS Behavior per Traffic Type

Traffic Type	CoS Behavior
Routed unicast	Unchanged
Bridged unicast	Unchanged
Routed multicast	Copied from 3 MSB of ToS
Bridged multicast with Layer 3 state for group	Copied from 3 MSB of ToS
Bridged multicast with no Layer 3 state for group	Unchanged

Note CoS behavior per traffic type is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

Prerequisites for Marking

Classification has the following prerequisites:

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

Guidelines and Limitations

Marking has the following configuration guidelines and limitations:

- **show** commands with the **internal** keyword are not supported.
- The **set qos-group** command can only be used in ingress policies.
- **Note** You can apply the marking instructions in a QoS policy map to ingress packets by attaching that QoS policy map to an interface. To select ingress, you specify the **input** keyword in the **service-policy** command.

  For more information, see the Attaching and Detaching a QoS Policy Action section.
- FEX host interfaces (HIF) are supported by the FEX QoS policy.

  Note FEX host interfaces are not supported on the Cisco Nexus 9508 switch (Cisco NX-OS Release 7.0(3)F3(3)).
- Control traffic, such as BPDUs, routing protocol packets, LACP/CDP/BFD, GOLD packets, glean traffic, and management traffic, are automatically classified into a control group based on a criteria. These packets are classified into qos-group 8 and have a strict absolute priority over other traffic. These packets are also given a dedicated buffer pool so that any congestion of data traffic does not affect control traffic. The control qos-group traffic classification cannot be modified.
- Span traffic automatically gets classified into qos-group 9 and is scheduled at absolute low priority.

Configuring Marking

You can combine one or more of the marking features in a policy map to control the setting of QoS values. You can then apply policies to either incoming or outgoing packets on an interface.

Note Do not press **Enter** after you use the **set** command and before you add the rest of the command. If you press **Enter** directly after entering the set keyword, you will be unable to continue to configure with the QoS configuration.

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 63, in addition to the standard DSCP values shown in the following table.

Table 28: Standard DSCP Values

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

Value	List of DSCP Values
default	Default dscp (000000)—decimal value 0
ef	EF dscp (101110)—decimal value 46

Note For more information about DSCP, see RFC 2475.

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map** [**type qos**] [**match-first**] *policy-map-name* **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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-name* | **class-default**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class1` `switch(config-pmap-c-qos)#`	Creates a reference to *class-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**set dscp** *dscp-value* **Example:** `switch(config-pmap-c-qos)# set dscp af31`	Sets the DSCP value to *dscp-value*. Standard values are shown in the previous Standard DSCP Values table. When the QoS policy is applied on the VLAN configuration level, the DSCP value derives the CoS value for bridged and routed traffic from the 3 most significant DSCP bits.

Example

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

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

Configuring IP Precedence Marking

You can set the value of the IP precedence field in bits 0–2 of the IPv4 type of service (ToS) field of the IP header.

Note The device rewrites the last 3 bits of the ToS field to 0 for packets that match this class.

Table 29: Precedence Values

Value	List of Precedence Values
0-7	IP precedence value
critical	Critical precedence (5)
flash	Flash precedence (3)
flash-override	Flash override precedence (4)
immediate	Immediate precedence (2)
internet	Internetwork control precedence (6)
network	Network control precedence (7)
priority	Priority precedence (1)
routine	Routine precedence (0)

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** [**type qos**] [**match-first**] *policy-map-name*
3. **class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*]
4. **set precedence** *precedence-value*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map** [**type qos**] [**match-first**] *policy-map-name* **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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.

	Command or Action	Purpose
Step 3	**class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class1` `switch(config-pmap-c-qos)#`	Creates a reference to *class-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before.
Step 4	**set precedence** *precedence-value* **Example:** `switch(config-pmap-c-qos)# set precedence 3`	Sets the IP precedence value to *precedence-value*. The value can range from 0 to 7. You can enter one of the values shown in the above Precedence Values table.

Example

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

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

Configuring CoS Marking

You can set the value of the CoS field in the high-order three bits of the VLAN ID Tag field in the IEEE 802.1Q header.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** [**type qos**] [**match-first**] [*qos-policy-map-name* | **qos-dynamic**]
3. **class** [**type qos**] {*class-map-name* | **class-default**} [**insert-before** *before-class-name*]
4. **set cos** *cos-value*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map** [**type qos**] [**match-first**] [*qos-policy-map-name* | **qos-dynamic**] **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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.
Step 3	**class** [**type qos**] {*class-map-name* | **class-default**} [**insert-before** *before-class-name*] **Example:**	Creates a reference to *class-map-name*, and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default**

	Command or Action	Purpose
	`switch(config-pmap-qos)# class class1` `switch(config-pmap-c-qos)#`	keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**set cos** *cos-value* **Example:** `switch(config-pmap-c-qos)# set cos 3` `switch(config-pmap-c-qos)#`	Sets the CoS value to *cos-value*. The value can range from 0 to 7.

Example

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

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

Configuring CoS Marking for FEX

Note The CoS Marking for FEX feature is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

You can mark traffic based on the class of service (CoS) for a FEX.

Before you begin

Before configuring the FEX, enable **feature-set fex**.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** [**type qos**] [**match-first**] [*qos-policy-map-name* | **qos-dynamic**]
3. **class** [**type qos**] {*class-map-name* | **class-default**} [**insert-before** *before-class-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map** [**type qos**] [**match-first**] [*qos-policy-map-name*	**qos-dynamic**] **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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.

	Command or Action	Purpose
Step 3	**class** [**type qos**] {*class-map-name* | **class-default**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class1` `switch(config-pmap-c-qos)#`	Creates a reference to *class-map-name*, and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.

Example

This example shows how to configure the CoS class-map configuration:

```
switch# conf t
switch(config)# policy-map type qos setpol
switch(config-pmap-qos)# class  cos6
switch(config-pmap-c-qos)# set qos-group 3
switch(config-pmap-qos)# class  cos3
switch(config-pmap-c-qos)# set qos-group 2
switch(config-pmap-qos)# class  cos1
switch(config-pmap-c-qos)# set qos-group 1
switch(config-pmap-qos)# class class-default
```

Configuring DSCP Port Marking

You can set the DSCP value for each class of traffic defined in a specified ingress policy map.

The default behavior of the device is to preserve the DSCP value or to trust DSCP. To make the port untrusted, change the DSCP value. Unless you configure a QoS policy and attach that policy to specified interfaces, the DSCP value is preserved.

Note

- You can attach only one policy type qos map to each interface in each direction.
- The DSCP value is trust on the Layer 3 port of a Cisco NX-OS device.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** [**type qos**] [**match-first**] [*policy-map-name*]
3. **class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*]
4. **set** *dscp-value*
5. **exit**
6. **class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*]
7. **set** *dscp-value*
8. **exit**
9. **class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*]
10. **set** *dscp-value*

11. **exit**
12. **interface ethernet** *slot/port*
13. **service-policy** [**type qos**] {**input** | **output**} {*policy-map-name*} [**no-stats**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map** [**type qos**] [**match-first**] [*policy-map-name*] **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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-name* | **class-default**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class1` `switch(config-pmap-c-qos)#`	Creates a reference to *class-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**set** *dscp-value* **Example:** `switch(config-pmap-c-qos)# set dscp af31`	Sets the DSCP value to dscp-value. Valid values are listed in the Standard DSCP Values table in the Configuring DSCP Marking section.
Step 5	**exit** **Example:** `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)#`	Returns to policy-map configuration mode.
Step 6	**class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class2` `switch(config-pmap-c-qos)#`	Creates a reference to *class-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 7	**set** *dscp-value* **Example:** `switch(config-pmap-c-qos)# set dscp af1`	Sets the DSCP value to dscp-value. Valid values are listed in the Standard DSCP Values table in the Configuring DSCP Marking section.
Step 8	**exit** **Example:** `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)#`	Returns to policy-map configuration mode.

	Command or Action	Purpose
Step 9	**class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class-default` `switch(config-pmap-c-qos)#`	Creates a reference to *class-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 10	**set** *dscp-value* **Example:** `switch(config-pmap-c-qos)# set dscp af22` `switch(config-pmap-c-qos)#`	Sets the DSCP value to dscp-value. Valid values are listed in the Standard DSCP Values table in the Configuring DSCP Marking section.
Step 11	**exit** **Example:** `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)#`	Returns to policy-map configuration mode.
Step 12	**interface ethernet** *slot/port* **Example:** `switch(config)# interface ethernet 1/1` `switch(config-if)#`	Enters interface mode to configure the Ethernet interface.
Step 13	**service-policy** [**type qos**] {**input** | **output**} {*policy-map-name*} [**no-stats**] **Example:** `switch(config-if)# service-policy input policy1`	Adds *policy-map-name* to the input packets of the interface. You can attach only one input policy and one output policy to an interface.

Example

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

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

Verifying the Marking Configuration

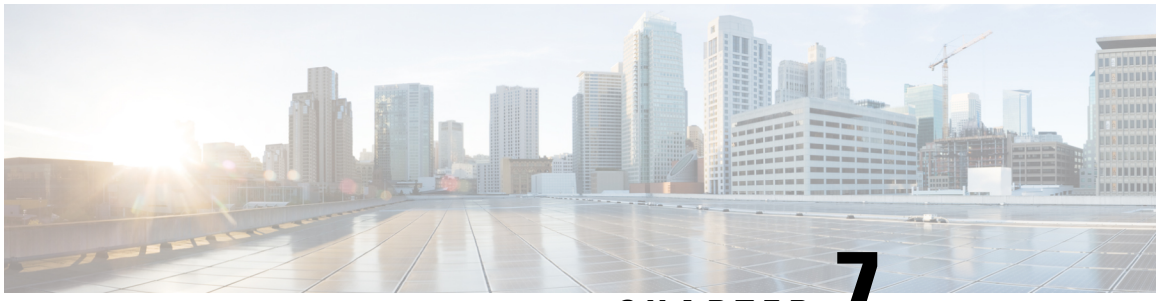
To display the marking configuration information, perform one of the following tasks:

Command	Purpose
show policy-map	Displays all policy maps.

Configuration Examples for Marking

The following example shows how to configure marking:

```
configure terminal
policy-map type qos untrust_dcsp
class class-default
set precedence 3
set qos-qroup 3
set dscp 0
```



CHAPTER 7

Configuring Policing

About Policing

Policing is the monitoring of the data rates for a particular class of traffic. When the data rate exceeds user-configured values, marking or dropping of packets occurs immediately. Policing does not buffer the traffic; therefore, the transmission delay is not affected. When traffic exceeds the data rate, you instruct the system to either drop the packets or mark QoS fields in them.

You can define single-rate and dual-rate policers.

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

You can configure only one action for each condition. For example, you might police for traffic in a class to conform to the data rate of 256000 bits per second, with up to 200 millisecond bursts. The system would apply the conform action to traffic that falls within this rate, and it would apply the violate action to traffic that exceeds this rate.

For more information about policers, see RFC 2697 and RFC 2698.

Prerequisites for Policing

Policing has the following prerequisites:

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

Guidelines and Limitations

Policing has the following configuration guidelines and limitations:

- Egress QoS policing is not supported on Cisco Nexus 9500 platform switches with the following line cards:
  - Cisco Nexus 9636C-R
  - Cisco Nexus 9636Q-R
  - Cisco Nexus 9636C-RX
  - Cisco Nexus 96136YC-R
- The egress RACL and egress QoS features are not supported on the Cisco Nexus 9508 switch (Cisco NX-OS Release 7.0(3)F3(3).
- Egress QoS policy statistics for CPU generated traffic are not supported on the following:
  - Cisco Nexus 9200, 9300-EX, and 9300-FX platform switches
  - Cisco Nexus 9500 platform switches with the following line cards:
    - Cisco Nexus 9732C-EX
    - Cisco Nexus 9736C-EX
    - Cisco Nexus 97160YC-EX
    - Cisco Nexus 9736C-FX
- **show** commands with the **internal** keyword are not supported.
- Each module polices independently, which might affect QoS features that are applied to traffic that is distributed across multiple modules. The following are examples of these QoS features:
  - Policers applied to a port channel interface.
  - Policers applied to a VLAN.
- All policers in the ingress direction must use the same mode.
- Policing only supports violated and non-violated statistics when using either double width or single width TCAM with e-qos-lite.
- Using the optional keyword, no-stats disables statistics and ensures that applicable policies are shared.
- You can only use the **set qos-group** command in ingress policies.
- When egress RACL and egress QoS are applied together, you can only enable statistics for one or the other, not both.
- Does not support egress QoS policies on ALE uplink ports on top-of-rack (TOR) platforms.
- When using egress QoS, Cisco recommends using the appropriate match criteria to match data traffic. (Avoid match criteria such as **permit ip any any**.)

- Beginning with Cisco NX-OS Release 7.0(3)I6(1), the Cisco Nexus 93108TC-EX, 93180YC-EX, and 93180LC-EX switches, and Cisco Nexus 97160YC-EX, 9732C-EX, 9736C-EX line cards support the Layer 2 and Layer 3 egress policer.
- Total number of policers that can be successfully attached in Egress direction is only half the size of qos-lite TCAM region.
- Beginning with Cisco NX-OS Release 7.0(3)I6(1), the Cisco Nexus 93180YC-EX and 93108TC-EX switches; and the Cisco Nexus 9736C-EX, 97160YC-EX, and 9732C-EX line cards do not support remark action for violated packets in the egress direction. They only support the drop action for violate in the egress direction.
- VLAN Egress QOS and Egress Qos on L2PO are not supported on Cisco Nexus 97160YC-EX, 9732C-EX, 9736C-EX line cards.
- Egress QoS policies are not supported on sub interfaces.
- For Cisco Nexus 9504 platform switches, egress QoS policies are not supported on the following:
  - Sub-interfaces
  - Physical interfaces with configured sub-interfaces
- Egress QoS policies are not supported on Cisco Nexus 9200 platform switches (For 7.0(3)I3(1)).
- Cisco Nexus 9200 platform switches support a 1-rate 2-color policer. A 2-rate 3-color policer is not supported on Cisco Nexus 9200 platform switches (For Cisco NX-OS Relese 7.0(3)I3(1)).
- QOS Ingress policies can be enabled on subinterfaces.
- Beginning with Cisco NX-OS Release 7.0(3)I6(1), the Cisco Nexus 93180YC-EX and 93108TC-EX switches; and the Cisco Nexus 9736C-EX, 97160YC-EX, and 9732C-EX line cards only support the 1R2C policing in the egress direction.
- Cisco Nexus 9200 platform Switches only support 1R2C policing in the ingress direction.

The following are guidelines and limitations for shared policers:

- When the shared policer is applied on interfaces or VLANs with member ports that are across different cores or instances, the rate becomes two times the configured **cir** rate. (Cisco NX-OS Release 7.0(3)F3(3) and later 7.0(3)F3(x) releases)

Configuring Policing

You can configure a single or dual-rate policer.

Configuring Ingress Policing

You can apply the policing instructions in a QoS policy map to ingress packets by attaching that QoS policy map to an interface. To select ingress, you specify the **input** keyword in the **service-policy** command. For more information on attaching and detaching a QoS policy action from an interface, see the "Using Modular QoS CLI" section.

Configuring Egress Policing

You can apply the policing instructions in a QoS policy map to ingress or egress packets by attaching that QoS policy map to an interface. To select ingress or egress, you specify the **input** keyword or the **output** keyword in the **service-policy** command.

Before you begin

- You must carve TCAM region for egress QoS before configuring policing.
- For more information about attaching and detaching a QoS policy action from an interface, see the "Using Modular QoS CLI" section.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** [**type qos**] [**match-first**] [*policy-map-name*]
3. **class** [**type qos**] {*class-map-name* | **class-default**} [**insert-before** *before-class-name*]
4. **police** [**cir**] {*committed-rate* [*data-rate*] | **percent** *cir-link-percent*} [**bc** *committed-burst-rate*] [**conform** {**transmit** | **set-prec-transmit** | **set-dscp-transmit** | **set-cos-transmit** | **set-qos-transmit**} [exceed { drop }**[violate** {**drop | set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **set-qos-transmit** }]]}
5. **exit**
6. **exit**
7. **show policy-map** [**type qos**] [*policy-map-name* | **qos-dynamic**]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map** [**type qos**] [**match-first**] [*policy-map-name*] **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class-default` `switch(config-pmap-c-qos)#`	Creates a reference to *class-map-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**police** [**cir**] {*committed-rate* [*data-rate*]	**percent** *cir-link-percent*} [**bc** *committed-burst-rate*] [**conform** {**transmit**	**set-prec-transmit**	**set-dscp-transmit**		Polices **cir** in bits or as a percentage of the link rate. The **conform** action is taken if the data rate is <= cir. The actions are described in the Policer Actions for Exceed or Violate table and the Policer Actions for Conform table. The data

	Command or Action	Purpose
	set-cos-transmit | **set-qos-transmit**} [exceed { drop }**[violate** {**drop** | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **set-qos-transmit** }]]} **Example:** `switch(config-pmap-qos)# policy-map type qos egressqos` `switch(config-pmap-qos)# class class-default` `switch(config-pmap-c-qos)# police [ cir] {committed-rate [data-rate] | percent cir-link-percent}` `[ bc committed-burst-rate][ conform { transmit | set-prec-transmit | set-dscp-transmit | set-cos-transmit | set-qos-transmit}] [ violate { drop}]}` `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)# exit` `switch(config)#`	rates and link speeds are described in the Data Rates for the police Command table and the Burst Sizes for the police Command table. See Configuring 1-Rate and 2-Rate, 2-Color and 3-Color Policing for more information. The following information describes the **drop** option for **violate**: • set-cos-transmit—Set dscp and send it. • set-prec-transmit—Set precedence and send it. • set-qos-transmit—Set qos-group and send it. **Note** For **cir** pps, the packet size is 64 bytes. So the pps to bps conversion is 64*8.
Step 5	**exit** **Example:** `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)#`	Exits policy-map class configuration mode and enters policy-map mode.
Step 6	**exit** **Example:** `switch(config-pmap-qos)# exit` `switch(config)#`	Exits policy-map mode and enters global configuration mode.
Step 7	**show policy-map** [**type qos**] [*policy-map-name* | **qos-dynamic**] **Example:** `switch(config)# show policy-map type qos egressqos` **Example:** `switch(config)# policy-map type qos egressqos` `class class-default` `police cir 10 mbs bc 200 ms conform transmit` `violate drop`	(Optional) Displays information about the configured policy map of type qos.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Configuring 1-Rate and 2-Rate, 2-Color and 3-Color Policing

The type of policer created by the device is based on a combination of the **police** command arguments described in the following Arguments to the police Command table.

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

Note A 1-rate 2-color policer with the violate markdown action is not supported.

Note Cisco Nexus 9200 Series switches only support 1-rate 2-color policing.

Table 30: Arguments to the police Command

Argument	Description
cir	Committed information rate, or desired bandwidth, specified as a bit rate or a percentage of the link rate. Although a value for cir is required, the argument itself is optional. The range of values is from 1 to 80000000000. The range of policing values is from 8000 to 80 Gbps.
percent	Rate as a percentage of the interface rate. The range of values is from 1 to 100 percent.
bc	Indication of how much the cir can be exceeded, either as a bit rate or an amount of time at cir. The default is 200 milliseconds of traffic at the configured rate. The default data rate units are bytes.
pir	Peak information rate, specified as a PIR bit rate or a percentage of the link rate. There is no default. The range of values is from 1 to 80000000000; the range of policing values is from 8000 bps to 480 Gbps. The range of percentage values is from 1 to 100 percent.
be	Indication of how much the pir can be exceeded, either as a bit rate or an amount of time at pir. When the bc value is not specified, the default is 200 milliseconds of traffic at the configured rate. The default data rate units are bytes. **Note** You must specify a value for pir before the device displays this argument.
conform	Single action to take if the traffic data rate is within bounds. The basic actions are transmit or one of the set commands listed in the following Policer Actions for Conform table. The default is transmit.
exceed	Single action to take if the traffic data rate is exceeded. The basic actions are drop or markdown. The default is drop.
violate	Single action to take if the traffic data rate violates the configured rate values. The basic actions are drop or markdown. The default is drop.

Although all the arguments in the above Arguments to the police Command table are optional, you must specify a value for **cir**. In this section, **cir** indicates its value but not necessarily the keyword itself. The combination of these arguments and the resulting policer types and actions are shown in the following Policer Types and Actions from Police Arguments Present table.

Table 31: Policer Types and Actions from Police Arguments Present

Police Arguments Present	Policer Type	Policer Action
cir, but not **pir**, **be**, or **violate**	1-rate, 2-color	<= **cir**, **conform**; else **violate**
cir and **pir**	2-rate, 3-color	<= **cir**, conform; <= **pir**, exceed; else **violate**

The policer actions that you can specify are described in the following Policer Actions for Exceed or Violate table and the following Policer Actions for Conform table.

Note Only **drop** and **transmit** actions are supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3) and later).

Table 32: Policer Actions for Exceed or Violate

Action	Description
drop	Drops the packet. This action is available only when the packet exceeds or violates the parameters.
set-cos-transmit	Sets CoS and transmits the packet.
set-dscp-transmit	Sets DSCP and transmits the packet.
set-prec-transmit	Sets precedence and transmits the packet.
set-qos-transmit	Sets qos-group and transmits the packet.

Table 33: Policer Actions for Conform

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

Note The policer can only drop or mark down packets that exceed or violate the specified parameters. For information on marking down packets, see the Configuring Marking, on page 63 section.

The data rates used in the **police** command are described in the following Data Rates for the police Command table.

Table 34: Data Rates for the police Command

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

Burst sizes used in the **police** command are described in the following Burst Sizes for the police Command table.

Table 35: Burst Sizes for the police Command

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

.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** [**type qos**] [**match-first**] [*policy-map-name*]
3. **class** [**type qos**] {*class-map-name* | **class-default**} [**insert-before** *before-class-name*]
4. **police** [**cir**] {*committed-rate* [*data-rate*] | **percent** *cir-link-percent*} [**bc** *committed-burst-rate* [*link-speed*]][**pir**] {*peak-rate* [*data-rate*] | **percent** *cir-link-percent*} [**be** *peak-burst-rate* [*link-speed*]] [**conform** {**transmit** | **set-prec-transmit** | **set-dscp-transmit** | **set-cos-transmit** | **set-qos-transmit**} [**exceed** {**drop**} [**violate** {**drop** | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **set-qos-transmit**}]]}
5. [violate {**drop** | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **set-qos-transmit**}]
6. **exit**
7. **exit**

8. **show policy-map** [**type qos**] [*policy-map-name* | **qos-dynamic**]
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map** [**type qos**] [**match-first**] [*policy-map-name*] **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class-default` `switch(config-pmap-c-qos)#`	Creates a reference to *class-map-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**police** [**cir**] {*committed-rate* [*data-rate*] | **percent** *cir-link-percent*} [**bc** *committed-burst-rate* [*link-speed*]][**pir**] {*peak-rate* [*data-rate*] | **percent** *cir-link-percent*} [**be** *peak-burst-rate* [*link-speed*]] [**conform** {**transmit** | **set-prec-transmit** | **set-dscp-transmit** | **set-cos-transmit** | **set-qos-transmit**} [**exceed** {**drop**} [**violate** {**drop** | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **set-qos-transmit**}]]}	Polices **cir** in bits or as a percentage of the link rate. The **conform** action is taken if the data rate is <= cir. If **be** and **pir** are not specified, all other traffic takes the **violate** action. If **be** or **violate** are specified, the **exceed** action is taken if the data rate <= **pir**, and the **violate** action is taken otherwise. The actions are described in the Policer Actions for Exceed or Violate table and the Policer Actions for Conform table. The data rates and link speeds are described in the Data Rates for the police Command table and the Burst Sizes for the police Command table.
Step 5	[violate {**drop** | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **set-qos-transmit**}]	**set-cos-transmit**—Set cos and send it. **set-dscp-transmit**—Set dscp and send it. **set-prec-transmit**—Set precedence and send it. **set-qos-transmit**—Set qos-group and send it.
Step 6	**exit** **Example:** `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)#`	Exits policy-map class configuration mode and enters policy-map mode.
Step 7	**exit** **Example:** `switch(config-pmap-qos)# exit` `switch(config)#`	Exits policy-map mode and enters global configuration mode.

	Command or Action	Purpose
Step 8	**show policy-map** [**type qos**] [*policy-map-name* | **qos-dynamic**] **Example:** `switch(config)# show policy-map`	(Optional) Displays information about all configured policy maps or a selected policy map of type qos.
Step 9	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to display the policy1 policy-map configuration:

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

Configuring Markdown Policing

Markdown policing is the setting of a QoS field in a packet when traffic exceeds or violates the policed data rates. You can configure markdown policing by using the set commands for policing action described in the Policer Actions for Exceed or Violate table and the Policer Actions for Conform table.

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

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** [**type qos**] [**match-first**] [*policy-map-name*]
3. **class** [**type qos**] {*class-name* | **class-default**} [**insert-before** *before-class-name*]
4. **police** [**cir**] {*committed-rate* [*data-rate*] | **percent** *cir-link-percent*} [[**bc** | **burst**] *burst-rate* [*link-speed*]] [[**be** | **peak-burst**] *peak-burst-rate* [*link-speed*]] [**conform** *conform-action* [**exceed** [**violate drop set dscp dscp table** *pir-markdown-map*]]}
5. **exit**
6. **exit**
7. **show policy-map** [**type qos**] [*policy-map-name*]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`switch# configure terminal` `switch(config)#`	
Step 2	**policy-map** [**type qos**] [**match-first**] [*policy-map-name*] **Example:** `switch(config)# policy-map policy1` `switch(config-pmap-qos)#`	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-name* | **class-default**} [**insert-before** *before-class-name*] **Example:** `switch(config-pmap-qos)# class class-default` `switch(config-pmap-c-qos)#`	Creates a reference to *class-name* and enters policy-map class configuration mode. The class is added to the end of the policy map unless **insert-before** is used to specify the class to insert before. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**police** [**cir**] {*committed-rate* [*data-rate*] | **percent** *cir-link-percent*} [[**bc** | **burst**] *burst-rate* [*link-speed*]] [[**be** | **peak-burst**] *peak-burst-rate* [*link-speed*]] [**conform** *conform-action* [**exceed** [**violate drop set dscp dscp table** *pir-markdown-map*]]}	Polices **cir** in bits or as a percentage of the link rate. The **conform** action is taken if the data rate is <= cir. If **be** and **pir** are not specified, all other traffic takes the **violate** action. If **be** or **violate** are specified, the **exceed** action is taken if the data rate <= **pir**, and the **violate** action is taken otherwise. The actions are described in the Policer Actions for Exceed or Violate table and the Policer Actions for Conform table. The data rates and link speeds are described in the Data Rates for the police Command table and the Burst Sizes for the police Command table.
Step 5	**exit** **Example:** `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)#`	Exits policy-map class configuration mode and enters policy-map mode.
Step 6	**exit** **Example:** `switch(config-pmap-qos)# exit` `switch(config)#`	Exits policy-map mode and enters global configuration mode.
Step 7	**show policy-map** [**type qos**] [*policy-map-name*] **Example:** `switch(config)# show policy-map`	(Optional) Displays information about all configured policy maps or a selected policy map of type qos.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Verifying the Policing Configuration

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

Command	Purpose
show policy-map	Displays information about policy maps and policing.

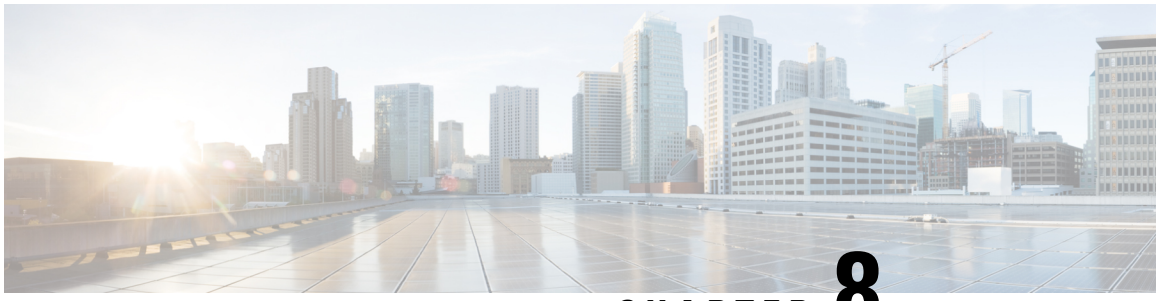
Configuration Examples for Policing

The following example shows how to configure policing for a 1-rate, 2-color policer:

```
configure terminal
  policy-map policy1
    class one_rate_2_color_policer
      police cir 256000 conform transmit violate drop
```

The following example shows how to configure policing for a 1-rate, 2-color policer with DSCP markdown:

```
configure terminal
  policy-map policy2
    class one_rate_2_color_policer_with_dscp_markdown
      police cir 256000 conform transmit violate drop
```



CHAPTER 8

Configuring Queuing and Scheduling

About Queuing and Scheduling

Traffic queuing is the ordering of packets and applies to both input and output of data. Device modules can support multiple queues, which you can use to control the sequencing of packets in different traffic classes. You can also set weighted random early detection (WRED) and taildrop thresholds. The device drops packets only when the configured thresholds are exceeded.

Traffic scheduling is the methodical output of packets at a desired frequency to accomplish a consistent flow of traffic. You can apply traffic scheduling to different traffic classes to weight the traffic by priority.

The queuing and scheduling processes allow you to control the bandwidth that is allocated to the traffic classes so that you achieve the desired trade-off between throughput and latency for your network.

Modifying Class Maps

System-defined queuing class maps are provided.

Note The provided system-defined queuing class maps cannot be modified.

Congestion Avoidance

You can use the following methods to proactively avoid traffic congestion on the device:

- Apply WRED to TCP or non-TCP traffic.
- Apply tail drop to TCP or non-TCP traffic.

Congestion Management

For egress packets, you can choose one of the following congestion management methods:

- Specify a bandwidth that allocates a minimum data rate to a queue.
- Impose a minimum and maximum data rate on a class of traffic so that excess packets are retained in a queue to shape the output rate.
- Allocate all data for a class of traffic to a priority queue. The device distributes the remaining bandwidth among the other queues.

For information about configuring congestion management, see the Configuring WRED on Egress Queues section.

Explicit Congestion Notification

ECN is an extension to WRED that marks packets instead of dropping them when the average queue length exceeds a specific threshold value. When configured with the WRED ECN feature, routers and end hosts use this marking as a signal that the network is congested to slow down sending packets.

Note The ECN feature is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3)).

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

Note On extended output queues (EOQ), the approximate fair-drop (AFD) feature for bandwidth management is always enabled. The WRED configuration is ignored on EOQs. The configuration for EOQs is based on the system queuing policy and not on the per port policy.

Traffic Shaping

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

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

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

Queue length thresholds are configured using the WRED configuration.

Note Traffic shaping is not supported on ALE enabled device 40G front panel ports. When traffic shaping is configured for the system level, the setting is ignored and no error message is displayed. When traffic shaping commands are configured for the port level, the setting is rejected and an error message is displayed.

Prerequisites for Queuing and Scheduling

Queuing and scheduling have the following prerequisites:

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

Guidelines and Limitations

Queuing and scheduling have the following configuration guidelines and limitations:

- **show** commands with the **internal** keyword are not supported.
- The device supports a system-level queuing policy, so all ports in the system are impacted when you configure the queuing policy.
- A type queuing policy can be attached to the system or to individual interfaces for input or output traffic.
- Changes are disruptive. The traffic passing through ports of the specified port type experience a brief period of traffic loss. All ports of the specified type are affected.

- Performance can be impacted. If one or more ports of the specified type do not have a queuing policy applied that defines the behavior for the new queue, the traffic mapping to that queue might experience performance degradation.
- Traffic shaping might increase the latency of packets due to queuing because it falls back to store-and-forward mode when packets are queued.
- Traffic shaping is not supported on the Cisco Nexus 9300 ALE 40G ports. For more information on ALE 40G uplink ports, see the Limitations for ALE 40G Uplink Ports on the Cisco Nexus 9000 Series Switches.
- When configuring priority for one class map queue (SPQ), you need to configure the priority for QoS group 3. When configuring priority for more than one class map queue, you need to configure the priority on the higher numbered QoS groups. In addition, the QoS groups need to be adjacent to each other. For example, if you want to have two SPQs, you have to configure the priority on QoS group 3 and on QoS group 2.
- For the following Cisco Nexus platform switches and line cards, the lowest value that the egress shaper can manage, per queue, is 100 Mbps:
  - Cisco Nexus 9200 platform switches
  - Cisco Nexus 9300-EX/FX/FX2 platform switches
  - Cisco Nexus 9700-EX/FX line cards

Buffer-boost

The buffer-boost feature enables the line card to use extra buffers. This capability is enabled by default on line cards such as the Cisco Nexus 9564PX.

- The command to enable the buffer-boost feature is **buffer-boost**.
- The command to disable the buffer-boost feature is **no buffer-boost**.

Generally, Cisco recommends not to disable the buffer-boost feature. However, disabling the buffer-boost is necessary when there is a need to port channel two different member ports from Cisco Nexus 9636PQ based line cards and Cisco Nexus 9564PX based line cards. However, Cisco does not recommend to port channel such a configuration between ACI capable leaf line cards and standalone line cards.

Note Line cards like the Cisco Nexus 9636PQ and similar, do not offer the buffer-boost feature.

Order of Resolution

The following describes the order of resolution for the pause buffer configuration and the queue-limit for a priority-group.

- Pause Buffer Configuration

  The pause buffer configuration is resolved in the following order:
  - Interface ingress queuing policy (if applied and pause buffer configuration specified for that class).
  - System ingress queuing policy (if applied and pause buffer configuration specified for that class).

- System network-QoS policy (if applied and pause buffer configuration specified for that class).
- Default values with regards to the speed of the port.

- Queue-limit for Priority-Group

  The queue-limit for a priority-group is resolved in the following order:

  - Interface ingress queuing policy (if applied and queue-limit configuration specified for that class).
  - System ingress queuing policy (if applied and queue-limit configuration specified for that class).
  - The **hardware qos ing-pg-share** configuration provided value.
  - System default value.

Ingress Queuing

The following are notes about ingress queuing:

- No default system ingress queuing policy exists.
- The ingress queuing policy is used to override the specified pause buffer configuration.
- When downgrading to an earlier release of Cisco NX-OS, all ingress queuing configurations have to be removed.
- The ingress queuing feature is supported only on platforms where priority flow control is supported.

Configuring Queuing and Scheduling

Queuing and scheduling are configured by creating policy maps of type queuing that you apply to an egress interface. You cannot modify system-defined class maps, which are used in policy maps to define the classes of traffic to which you want to apply policies.

System-defined class maps match based on QoS groups that can be customized using a type qos policy. By default, there is no type QoS policy and all traffic matches to qos-group 0. One consequence is that all traffic will hit the system-defined default-class of type network-qos and type queuing (assigns 100% bandwidth to qos-group 0). Since system-defined classes of type queuing and type network-qos are predefined to match based on distinct qos-groups and cannot be modified, the way to ensure that traffic hits a given type queuing/network-qos class is to configure a type qos policy that sets the corresponding qos-group for that traffic. For traffic classified into a system-defined class map matching on a qos-group other than 0, create a type QoS policy that sets the QoS groups. Once the traffic has been mapped, it will be subject to the default type network-qos and type queuing policies that operate on the non-default qos-group X (X !=0). You may need to further customize those type queuing and type network-qos policies in order to ensure the desired actions (e.g. re-allocate some bandwidth). For more information on setting the qos-group, see “Example of set qos-groups” in the Using Modular QoS CLI chapter.

For information about configuring policy maps and class maps, see the Using Modular QoS CLI chapter.

You can configure the congestion-avoidance features, which include tail drop and WRED, in any queue.

You can configure one of the egress congestion management features, such as priority, traffic shaping, and bandwidth in output queues.

Note WRED is not supported on ALE enabled device front panel 40G uplink ports. When WRED is configured for the system level, the setting is ignored and no error message is displayed. When WRED is configured at the port level, the setting is rejected and an error message displays.

The system-defined policy map, default-out-policy, is attached to all ports to which you do not apply a queuing policy map. The default policy maps cannot be configured.

Configuring Type Queuing Policies

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

Note Ingress queuing policy is used to configure pause buffer thresholds. For more details, see the About Priority Flow Control section.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** *policy-name*
3. **class type queuing** *class-name*
4. **priority**
5. **no priority**
6. **shape** {**kbps** | **mbps** | **gbps**} *burst size* **min** *minimum bandwidth*
7. **bandwidth percent** *percentage*
8. **no bandwidth percent** *percentage*
9. **priority level** *level*
10. **queue-limit** *queue size* [**dynamic** *dynamic threshold*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**policy-map type queuing** *policy-name*	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.
Step 3	**class type queuing** *class-name*	Associates a class map with the policy map, and enters configuration mode for the specified system class.
Step 4	**priority**	Specifies that traffic in this class is mapped to a strict priority queue.

	Command or Action	Purpose
Step 5	**no priority**	(Optional) Removes the strict priority queuing from the traffic in this class.
Step 6	**shape** {**kbps** | **mbps** | **gbps**} *burst size* **min** *minimum bandwidth*	Specifies the burst size and minimum guaranteed bandwidth for this queue.
Step 7	**bandwidth percent** *percentage*	Assigns a weight to the class. The class will receive the assigned percentage of interface bandwidth if there are no strict-priority queues. If there are strict-priority queues, however, the strict-priority queues receive their share of the bandwidth first. The remaining bandwidth is shared in a weighted manner among the class configured with a bandwidth percent. For example, if strict-priority queues take 90 percent of the bandwidth, and you configure 75 percent for a class, the class will receive 75 percent of the remaining 10 percent of the bandwidth. **Note** Before you can successfully allocate bandwidth to the class, you must first reduce the default bandwidth configuration on class-default and class-fcoe.
Step 8	**no bandwidth percent** *percentage*	(Optional) Removes the bandwidth specification from this class.
Step 9	**priority level** *level*	(Optional) Specifies the strict priority levels for the Cisco Nexus 9000 Series switches. These levels can be from 1 to 7.
Step 10	**queue-limit** *queue size* [**dynamic** *dynamic threshold*]	(Optional) Specifies either the static or dynamic shared limit available to the queue for Cisco Nexus 9000 Series switches. The static queue limit defines the fixed size to which the queue can grow. The dynamic queue limit allows the queue's threshold size to be decided depending on the number of free cells available, in terms of the alpha value. **Note** Cisco Nexus 9200 Series switches only support a class level dynamic threshold configuration with respect to the alpha value. This means that all ports in a class share the same alpha value.

Configuring Congestion Avoidance

You can configure congestion avoidance with tail drop or WRED features. Both features can be used in egress policy maps.

Note WRED and tail drop cannot be configured in the same class.

Configuring Tail Drop on Egress Queues

You can configure tail drop on egress queues by setting thresholds. The device drops any packets that exceed the thresholds. You can specify a threshold based on the queue size or buffer memory that is used by the queue.

SUMMARY STEPS

1. **configure terminal**
2. **hardware qos q-noise percent** *value*
3. **policy-map** [**type queuing**] [**match-first**] [*policy-map-name*]
4. **class type queuing** *class-name*
5. **queue-limit** {*queue-size* [**bytes** | **kbytes** | **mbytes**] | **dynamic** *value*}
6. (Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.
7. **show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**hardware qos q-noise percent** *value* **Example:** `switch(config)# hardware qos q-noise percent 30`	Tunes the random noise parameter. The default value is 20 percent. This command is supported for Cisco Nexus 9200 and 9300-EX Series switches beginning with Cisco NX-OS Release 7.0(3)I4(4).
Step 3	**policy-map** [**type queuing**] [**match-first**] [*policy-map-name*] **Example:** `switch(config)# policy-map type queuing shape_queues` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 4	**class type queuing** *class-name* **Example:** `switch(config-pmap-que)# class type queuing c-out-q1` `switch(config-pmap-c-que)#`	Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table.

	Command or Action	Purpose
Step 5	**queue-limit** {*queue-size* [**bytes** | **kbytes** | **mbytes**] | **dynamic** *value*} **Example:** `switch(config-pmap-c-que)# queue-limit 1000 mbytes`	Assigns a tail drop threshold based on the queue size in bytes, kilobytes, or megabytes or allows the queue's threshold size to be determined dynamically depending on the number of free cells available. The device drops packets that exceed the specified threshold. The valid values for byte-based queue size are from 1 to 83886080. The valid values for dynamic queue size are from 0 to 10 as follows: For example, if you configure a dynamic queue size of 6, then the alpha value is ½. If you configure a dynamic queue size of 7, then the alpha value is 1. To calculate the queue-limit consider the following: queue-limit = (alpha/(1 + alpha)) x total buffers For example, if you configure a queue-limit with a dynamic queue size of 7, then the queue-limit can grow up to (1/(1+1)) x total buffers. This means that queue-limit = ½ x total buffers. **Note** Setting the threshold on ALE enabled devices is only supported for the system level. It is not supported for the port level.
Step 6	(Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.	
Step 7	**show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]] **Example:** `switch(config-pmap-c-que)# show policy-map type queuing shape_queues`	(Optional) Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Configuring WRED on Egress Queues

You can configure WRED on egress queues to set minimum and maximum packet drop thresholds. The frequency of dropped packets increases as the queue size exceeds the minimum threshold. When the maximum threshold is exceeded, all packets for the queue are dropped.

Note WRED and tail drop cannot be configured in the same class.

Note AFD and WRED cannot be applied at the same time. Only one can be used in a system.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** {[**match-first**] *policy-map-name*}
3. **class type queuing** *class-name*
4. **random-detect** [**minimum-threshold** *min-threshold* {**packets** | **bytes** | **kbytes** | **mbytes**} **maximum-threshold** *max-threshold* {**packets** | **bytes** | **kbytes** | **mbytes**} **drop-probability** *value* **weight** *value*] [**threshold** {**burst-optimized** | **mesh-optimized**}] [**ecn** | **non-ecn**]
5. (Optional) Repeat Steps 3 and 4 to configure WRED for other queuing classes.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map type queuing** {[**match-first**] *policy-map-name*} **Example:** `switch(config)# policy-map type queuing p1` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class type queuing** *class-name* **Example:** `switch(config-pmap-que)# class type queuing c-out-q1` `switch(config-pmap-c-que)#`	Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table.
Step 4	**random-detect** [**minimum-threshold** *min-threshold* {**packets** | **bytes** | **kbytes** | **mbytes**} **maximum-threshold** *max-threshold* {**packets** | **bytes** | **kbytes** | **mbytes**} **drop-probability** *value* **weight** *value*] [**threshold** {**burst-optimized** | **mesh-optimized**}] [**ecn** | **non-ecn**] **Example:** `switch(config-pmap-c-que)# random-detect minimum-threshold 10 mbytes maximum-threshold 20 mbytes`	Configures WRED on the specified queuing class. You can specify minimum and maximum thresholds used to drop packets from the queue. You can configure these thresholds by the number of packets, bytes, kilobytes, or megabytes. The minimum and maximum thresholds must be of the same type. The thresholds are from 1 to 52428800. **Note** The minimum-threshold and maximum-threshold parameters are not supported on the Cisco Nexus 9300 platform switches and Cisco Nexus 9564TX and 9564PX line cards. When random-detect is configured under policy-map the default thresholds and drop probabilities are as following:

	Command or Action	Purpose
		a. On newer platforms, the threshold is 0 and then the drop probabilities would be enforced irrespective of buffer utilization. **b.** On older platforms, the threshold is min 100KB, max 120KB. The drop probabilities are consistently 10% and 90% for burst-optimized and mesh-optimized respectively on all platforms
Step 5	(Optional) Repeat Steps 3 and 4 to configure WRED for other queuing classes.	

Configuring Congestion Management

You can configure only one of the following congestion management methods in a policy map:

- Allocate a minimum data rate to a queue by using the **bandwidth** and **bandwidth remaining** commands.
- Allocate all data for a class of traffic to a priority queue by using the **priority** command. You can use the **bandwidth remaining** command to distribute remaining traffic among the nonpriority queues. By default, the system evenly distributes the remaining bandwidth among the nonpriority queues.
- Allocate a minimum and maximum data rate to a queue by using the **shape** command.

In addition to the congestion management feature that you choose, you can configure one of the following queue features in each class of a policy map:

- Tail drop thresholds based on the queue size and the queue limit usage. For more information, see Configuring Tail Drop on Egress Queues, on page 92.
- WRED for preferential packet drops. For more information, see the Configuring WRED on Egress Queues section.

Note WRED is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3).

Configuring Bandwidth and Bandwidth Remaining

You can configure the bandwidth and bandwidth remaining on the egress queue to allocate a minimum percentage of the interface bandwidth to a queue.

Note When a guaranteed bandwidth is configured, the priority queue must be disabled in the same policy map.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** {[**match-first**] *policy-map-name*}
3. **class type queuing***class-name*
4. Assign a minimum rate of the interface bandwidth or assign the percentage of the bandwidth that remains:
   - Bandwidth percent:

     bandwidth {**percent** *percent*}
   - Bandwidth remaining percent:

     bandwidth remaining percent *percent*
5. (Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.
6. exit
7. **show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map type queuing** {[**match-first**] *policy-map-name*} **Example:** `switch(config)# policy-map type queuing shape_queues` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class type queuing***class-name* **Example:** `switch(config-pmap-que)# class type queuing c-out-q1` `switch(config-pmap-c-que)#`	Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table.
Step 4	Assign a minimum rate of the interface bandwidth or assign the percentage of the bandwidth that remains: • Bandwidth percent: **bandwidth** {**percent** *percent*} • Bandwidth remaining percent: **bandwidth remaining percent** *percent* **Example:** • Bandwidth percent:	• Bandwidth percent: Assigns a minimum rate of the interface bandwidth to an output queue as the percentage of the underlying interface link rate. The range is from 0 to 100. The example shows how to set the bandwidth to a minimum of 25 percent of the underlying link rate. • Bandwidth remaining percent: Assigns the percentage of the bandwidth that remains to this queue. The range is from 0 to 100.

	Command or Action	Purpose
	`switch(config-pmap-c-que)# bandwidth percent 25` • Bandwidth remaining percent: `switch(config-pmap-c-que)# bandwidth remaining percent 25`	The example shows how to set the bandwidth for this queue to 25 percent of the remaining bandwidth.
Step 5	(Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.	
Step 6	exit **Example:** `switch(config-cmap-que)# exit` `switch(config)#`	Exits policy-map queue mode and enters global configuration mode.
Step 7	**show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]] **Example:** `switch(config-pmap-c-que)# show policy-map type queuing shape_queues`	(Optional) Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Configuring Bandwidth and Bandwidth Remaining for FEX

You can configure the bandwidth and bandwidth remaining on the ingress and egress queue to allocate a minimum percentage of the interface bandwidth to a queue.

Note When a guaranteed bandwidth is configured, the priority queue must be disabled in the same policy map.

Before you begin

Before configuring the FEX, enable **feature-set fex**.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** {[**match-first**] *policy-map-name*}
3. **class type queuing***class-name*
4. Assign a minimum rate of the interface bandwidth or assign the percentage of the bandwidth that remains:
   - Bandwidth percent:

     bandwidth {**percent** *percent*}

- Bandwidth remaining percent:

  bandwidth remaining percent *percent*

5. (Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.
6. exit
7. **show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map type queuing** {[**match-first**] *policy-map-name*} **Example:** `switch(config)# policy-map type queuing shape_queues` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class type queuing***class-name* **Example:** `switch(config-pmap-que)# class type queuing c-out-q1` `switch(config-pmap-c-que)#`	Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table.
Step 4	Assign a minimum rate of the interface bandwidth or assign the percentage of the bandwidth that remains: • Bandwidth percent: **bandwidth** {**percent** *percent*} • Bandwidth remaining percent: **bandwidth remaining percent** *percent* **Example:** • Bandwidth percent: `switch(config-pmap-c-que)# bandwidth percent 25` • Bandwidth remaining percent: `switch(config-pmap-c-que)# bandwidth remaining percent 25`	• Bandwidth percent: Assigns a minimum rate of the interface bandwidth to an output queue as the percentage of the underlying interface link rate. The range is from 0 to 100. The example shows how to set the bandwidth to a minimum of 25 percent of the underlying link rate. • Bandwidth remaining percent: Assigns the percentage of the bandwidth that remains to this queue. The range is from 0 to 100. The example shows how to set the bandwidth for this queue to 25 percent of the remaining bandwidth.
Step 5	(Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.	

	Command or Action	Purpose
Step 6	exit **Example:** `switch(config-cmap-que)# exit` `switch(config)#`	Exits policy-map queue mode and enters global configuration mode.
Step 7	**show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]] **Example:** `switch(config-pmap-c-que)# show policy-map type queuing shape_queues`	(Optional) Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to configure the interface bandwidth:

```
switch(config)# policy-map type queuing inq
switch(config-pmap-que)# class type queuing c-in-q3
switch(config-pmap-c-que)# bandwidth percent 30
switch(config-pmap-que)# class type queuing c-in-q2
switch(config-pmap-c-que)# bandwidth percent 20
switch(config-pmap-que)# class type queuing c-in-q1
switch(config-pmap-c-que)# bandwidth percent 10
switch(config-pmap-que)# class type queuing c-in-q-default
switch(config-pmap-c-que)# bandwidth percent 40
```

Configuring Priority

If you do not specify the priority, the system-defined egress pq queues behave as normal queues. For information on the system-defined type queuing class maps, see the "Using Modular QoS CLI" section.

You can configure only one level of priority on an egress priority queue. You use the system-defined priority queue class for the type of module to which you want to apply the policy map.

For the nonpriority queues, you can configure how much of the remaining bandwidth to assign to each queue. By default, the device evenly distributes the remaining bandwidth among the nonpriority queues.

Note When a priority queue is configured, the other queues can only use the remaining bandwidth in the same policy map.

Note When configuring priority for one class map queue (SPQ), you need to configure the priority for QoS group 3. When configuring priority for more than one class map queue, you need to configure the priority on the higher numbered QoS groups. In addition, the QoS groups need to be adjacent to each other. For example, if you want to have two SPQs, you have to configure the priority on QoS group 3 and on QoS group 2.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** {[**match-first**] *policy-map-name*}
3. **class type queuing** *class-name*
4. **priority** [**level** *value*]
5. **class type queuing***class-name*
6. **bandwidth remaining percent** *percent*
7. (Optional) Repeat Steps 5 to 6 to assign the remaining bandwidth for the other nonpriority queues.
8. exit
9. **show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]]
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map type queuing** {[**match-first**] *policy-map-name*} **Example:** `switch(config)# policy-map type queuing priority_queue1` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class type queuing** *class-name* **Example:** `switch(config-pmap-que)# class type queuing c-out-q1` `switch(config-pmap-c-que)#`	Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table.
Step 4	**priority** [**level** *value*] **Example:** `switch(config-pmap-c-que)# priority`	Selects this queue as a priority queue. Only one priority level is supported.
Step 5	**class type queuing***class-name* **Example:**	(Optional) Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing

	Command or Action	Purpose
	`switch(config-pmap-que)# class type queuing c-out-q2` `switch(config-pmap-c-que)#`	names are listed in the previous System-Defined Type queuing Class Maps table. Choose a nonpriority queue where you want to configure the remaining bandwidth. By default, the system evenly distributes the remaining bandwidth among the nonpriority queues.
Step 6	**bandwidth remaining percent** *percent* **Example:** `switch(config-pmap-c-que)# bandwidth remaining percent 25`	(Optional) Assigns the percent of the bandwidth that remains to this queue. The range is from 0 to 100.
Step 7	(Optional) Repeat Steps 5 to 6 to assign the remaining bandwidth for the other nonpriority queues.	
Step 8	exit **Example:** `switch(config-cmap-que)# exit` `switch(config)#`	Exits policy-map queue mode and enters global configuration mode.
Step 9	**show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]] **Example:** `switch(config)# show policy-map type queuing priority_queue1`	(Optional) Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.
Step 10	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Configuring Priority for FEX

Note Priority for FEX is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3).

If you do not specify the priority, the system-defined egress pq queues behave as normal queues. For information on the system-defined type queuing class maps, see the "Using Modular QoS CLI" section.

You can configure only one level of priority on an egress priority queue. You use the system-defined priority queue class for the type of module to which you want to apply the policy map.

For the nonpriority queues, you can configure how much of the remaining bandwidth to assign to each queue. By default, the device evenly distributes the remaining bandwidth among the non-priority queues.

Note When a priority queue is configured, the other queues can only use the remaining bandwidth in the same policy map.

Note When configuring priority for one class map queue (SPQ), you need to configure the priority for QoS group 3. When configuring priority for more than one class map queue, you need to configure the priority on the higher numbered QoS groups. In addition, the QoS groups need to be adjacent to each other. For example, if you want to have two SPQs, you have to configure the priority on QoS group 3 and on QoS group 2.

Before you begin

Before configuring the FEX, enable **feature-set fex**.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** {[**match-first**] *policy-map-name*}
3. **class type queuing** *class-name*
4. **priority** [**level** *value*]
5. **class type queuing** *class-name*
6. **bandwidth remaining percent** *percent*
7. (Optional) Repeat Steps 5 to 6 to assign the remaining bandwidth for the other nonpriority queues.
8. exit
9. **show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]]
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map type queuing** {[**match-first**] *policy-map-name*} **Example:** `switch(config)# policy-map type queuing priority_queue1` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class type queuing** *class-name* **Example:**	Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table.

	Command or Action	Purpose
	`switch(config-pmap-que)# class type queuing c-out-q3` `switch(config-pmap-c-que)#`	
Step 4	**priority** [**level** *value*] **Example:** `switch(config-pmap-c-que)# priority`	Selects this queue as a priority queue. Only one priority level is supported. **Note** FEX QoS priority is supported only on the c-out-q3 class map.
Step 5	**class type queuing** *class-name* **Example:** `switch(config-pmap-que)# class type queuing c-out-q3` `switch(config-pmap-c-que)#`	(Optional) Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table. Choose a nonpriority queue where you want to configure the remaining bandwidth. By default, the system evenly distributes the remaining bandwidth among the nonpriority queues.
Step 6	**bandwidth remaining percent** *percent* **Example:** `switch(config-pmap-c-que)# bandwidth remaining percent 25`	(Optional) Assigns the percent of the bandwidth that remains to this queue. The range is from 0 to 100.
Step 7	(Optional) Repeat Steps 5 to 6 to assign the remaining bandwidth for the other nonpriority queues.	
Step 8	exit **Example:** `switch(config-cmap-que)# exit` `switch(config)#`	Exits policy-map queue mode and enters global configuration mode.
Step 9	**show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]] **Example:** `switch(config)# show policy-map type queuing priority_queue1`	(Optional) Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.
Step 10	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Example

This example shows how to configure the level of priority:

```
switch(config)# policy-map type queuing inq_pri
switch(config-pmap-que)# class type queuing c-in-q3
```

```
switch(config-pmap-c-que)# priority
switch(config-pmap-que)# class type queuing c-in-q2
switch(config-pmap-c-que)# bandwidth remaining percent 20
switch(config-pmap-que)# class type queuing c-in-q1
switch(config-pmap-c-que)# bandwidth remaining percent 40
switch(config-pmap-que)# class type queuing c-in-q-default
switch(config-pmap-c-que)# bandwidth remaining percent 40
```

Configuring Traffic Shaping

You can configure traffic shaping on an egress queue to impose a minimum and maximum rate on it.

Note Configuring traffic shaping for a queue is independent of priority or bandwidth in the same policy map.

Note The system queuing policy is applied to both internal and front panel ports. When traffic shaping is enabled on the system queuing policy, traffic shaping is also applied to the internal ports. As a best practice, do not enable traffic shaping on the system queuing policy.

Note Traffic shaping is not supported on the Cisco Nexus 9300 40G ports.

Note The lowest value that the egress shaper can manage, per queue, is 100 Mbps on Cisco Nexus 9200 series, 9300-EX/FX/FX2, and 9700-EX/FX switches.

Before you begin

Configure random detection minimum and maximum thresholds for packets.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** {[**match-first**] *policy-map-name*}
3. **class type queuing** *class-name*
4. **shape min** *value* {**bps** | **gbps** | **kbps** | **mbps** | **pps**} **max** *value* {**bps** | **gbps** | **kbps** | **mbps** | **pps**}
5. (Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.
6. **show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**policy-map type queuing** {[**match-first**] *policy-map-name*} **Example:** `switch(config)# policy-map type queuing shape_queues` `switch(config-pmap-que)#`	Configures the policy map of type queuing and then enters policy-map mode for the policy-map name you specify. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class type queuing** *class-name* **Example:** `switch(config)# class type queuing c-out-q-default` `switch(config-pmap-c-que)#`	Configures the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the previous System-Defined Type queuing Class Maps table.
Step 4	**shape min** *value* {**bps** | **gbps** | **kbps** | **mbps** | **pps**} **max** *value* {**bps** | **gbps** | **kbps** | **mbps** | **pps**} **Example:** `switch(config-pmap-c-que)# shape min 10 bps max 100 bps`	Assigns a minimum and maximum bit rate on an output queue. The default bit rate is in bits per second (bps). The example shows how to shape traffic to a minimum rate of 10 bits per second (bps) and a maximum rate of 100 bps. **Note** Most scenarios where traffic shaping is needed requires the configuration of only the max shaper value. For instance, if you want traffic shaped and limited to a maximum desired rate, configure the min shaper value as 0 and the max shaper value as the maximum desired rate. You should only configure the min shaper value for specific scenarios where a guaranteed rate is desired. For instance, if you want traffic to have a guaranteed rate, configure the min shaper value as the guaranteed rate and the max value as something greater than guaranteed rate (or the maximum of the port speed rate).
Step 5	(Optional) Repeat Steps 3 and 4 to assign tail drop thresholds for other queue classes.	
Step 6	**show policy-map** [**type queuing** [*policy-map-name* | **default-out-policy**]] **Example:** `switch(config)# show policy-map type queuing shape_queues`	(Optional) Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.

	Command or Action	Purpose
Step 7	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Applying a Queuing Policy on a System

You apply a queuing policy globally on a system.

SUMMARY STEPS

1. **configure terminal**
2. **system qos**
3. **service-policy type queuing output** {*policy-map-name* | **default-out-policy**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**system qos** **Example:** `switch (config)# system qos` `switch (config-sys-qos)#`	Enters system qos mode.
Step 3	**service-policy type queuing output** {*policy-map-name*	**default-out-policy**} **Example:** `switch (config-sys-qos)# service-policy type queuing map1`	Adds the policy map to the input or output packets of system. **Note** The **output** keyword specifies that this policy map should be applied to traffic transmitted from an interface. **Note** To restore the system to the default queuing service policy, use the **no** form of this command.

Verifying the Queuing and Scheduling Configuration

Use the following commands to verify the queuing and scheduling configuration:

Command	Purpose
show class-map [**type queuing** [*class-name*]]	Displays information about all configured class maps, all class maps of type queuing, or a selected class map of type queuing.
show policy-map [**type queuing** [*policy-map-name* | **default-out-policy**]]	Displays information about all configured policy maps, all policy maps of type queuing, a selected policy map of type queuing, or the default output queuing policy.
show policy-map system	Displays information about all configured policy maps on the system.

Controlling the QoS Shared Buffer

The QoS buffer provides support per port/queue and shared space. You can control the QoS buffer that is shared by all flows by disabling or restricting reservations.

The **hardware qos min-buffer** command is used to control the QoS shared buffer.

hardware qos min-buffer [**all**|**default**|**none**]	• **all** Current behavior where all reservations are enabled ON). • **default** Enables reservations only for qos-group-0. • **none** Disables reservations for all qos-groups.

The **show hardware qos min-buffer** command is used to display the current buffer configuration.

Monitoring the QoS Packet Buffer

The Cisco Nexus 9000 Series device has a 12-MB buffer memory that divides into a dedicated per port and dynamic shared memory. Each front-panel port has four unicast queues and four multicast queues in egress. In the scenario of burst or congestion, each egress port consumes buffers from the dynamic shared memory.

You can display the real-time and peak status of the shared buffer per port. All counters are displayed in terms of the number of cells. Each cell is 208 bytes in size. You can also display the global level buffer consumption in terms of consumption and available number of cells.

Note Monitoring the shared buffer on ALE enabled devices is not supported for the port level.

Note In the examples shown in this section, the port numbers are Broadcom ASIC ports.

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

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

This example shows how to set a buffer utilization threshold for a specific module:

```
switch(config)# hardware profile buffer info port-threshold module 1 threshold 10
Port threshold changed successfully
```

Note The buffer threshold feature is not enabled for ports if they have a no-drop class configured (PFC).

Note The configured threshold buffer count is checked every 5 seconds against all the buffers used by that port across all the queues of that port.

Note You can configure the threshold percentage configuration for all modules or for a specific module, which is applied to all ports. The default threshold value is 90% of the switch cell count of shared pool SP-0. This configuration applies to both Ethernet (front panel) and internal (HG) ports.

Note The buffer threshold feature is not supported for ACI capable device ports.

This example shows how to display the interface hardware mappings:

```
eor15# show interface hardware-mappings
Legends:
       SMod  - Source Mod. 0 is N/A
       Unit  - Unit on which port resides. N/A for port channels
       HPort - Hardware Port Number or Hardware Trunk Id:
       FPort - Fabric facing port number. 255 means N/A
       NPort - Front panel port number
       VPort - Virtual Port Number. -1 means N/A

-------------------------------------------------------------------
Name       Ifindex  Smod Unit HPort FPort NPort VPort
-------------------------------------------------------------------
Eth2/1     1a080000 4    0    13    255   0     -1
Eth2/2     1a080200 4    0    14    255   1     -1
Eth2/3     1a080400 4    0    15    255   2     -1
Eth2/4     1a080600 4    0    16    255   3     -1
Eth2/5     1a080800 4    0    17    255   4     -1
```

```
Eth2/6     1a080a00 4    0    18    255   5     -1
Eth2/7     1a080c00 4    0    19    255   6     -1
Eth2/8     1a080e00 4    0    20    255   7     -1
Eth2/9     1a081000 4    0    21    255   8     -1
Eth2/10    1a081200 4    0    22    255   9     -1
Eth2/11    1a081400 4    0    23    255   10    -1
Eth2/12    1a081600 4    0    24    255   11    -1
Eth2/13    1a081800 4    0    25    255   12    -1
Eth2/14    1a081a00 4    0    26    255   13    -1
Eth2/15    1a081c00 4    0    27    255   14    -1
Eth2/16    1a081e00 4    0    28    255   15    -1
Eth2/17    1a082000 4    0    29    255   16    -1
Eth2/18    1a082200 4    0    30    255   17    -1
Eth2/19    1a082400 4    0    31    255   18    -1
Eth2/20    1a082600 4    0    32    255   19    -1
Eth2/21    1a082800 4    0    33    255   20    -1
Eth2/22    1a082a00 4    0    34    255   21    -1
Eth2/23    1a082c00 4    0    35    255   22    -1
Eth2/24    1a082e00 4    0    36    255   23    -1
```

Configuration Examples for Queuing and Scheduling

In this section, you can find examples of configuring queuing and scheduling.

Note The default system classes type queuing match based on qos-group (by default all traffic matches to qos-group 0, and this default queue gets 100% bandwidth). Create a type QoS policy that first sets the qos-group in order to drive the correct matching for the type queuing classes and policies.

Example: Configuring WRED on Egress Queues

The following example shows how to configure the WRED feature on an egress queue:

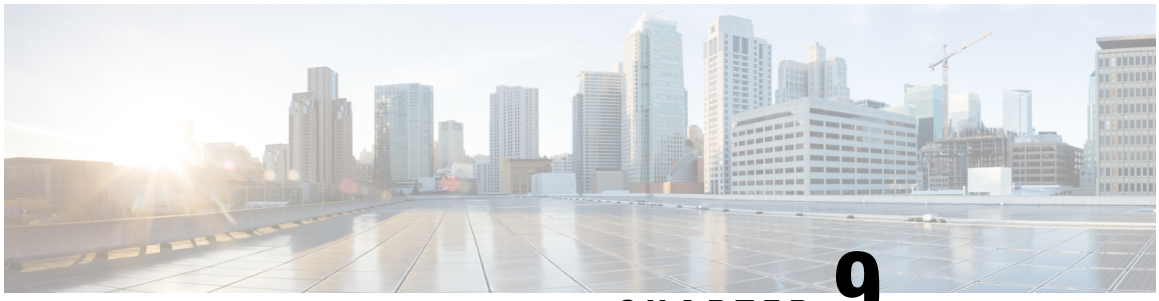
```
configure terminal
  class-map type queuing match-any c-out-q1
    match qos-group 1
  class-map type queuing match-any c-out-q2
    match qos-group 1
  policy-map type queuing wred
    class type queuing c-out-q1
      random-detect minimum-threshold 10 bytes maximum-threshold 1000 bytes
    class type queuing c-out-q2
      random-detect threshold burst-optimized ecn
```

Example: Configuring Traffic Shaping

The following example shows how to configure traffic shaping using 1000 packets per second (pps)::

```
configure terminal
  class-map type queuing match-any c-out-q1
    match qos-group 1
  class-map type queuing match-any c-out-q2
    match qos-group 1
policy-map type queuing pqu
 class type queuing c-out-q1
```

```
  shape min 100 pps max 500 pps
 class type queuing c-out-q2
  shape min 200 pps max 1000 pps
show policy-map type queuing pqu
```



CHAPTER 9

Configuring Network QoS

About Network QoS

The network QoS policy defines the characteristics of QoS properties network wide. With a network QoS policy, you can configure the following:

- Pause behavior—You can decide whether a QoS group requires the lossless behavior. The lossless behavior is provided by using a priority flow control (PFC) mechanism that prevents packet loss during congestion. You can configure drop (frames with this value that can be dropped) and no drop (frames with this value that cannot be dropped). For the drop and no drop configuration, you also need to enable PFC per port. For more information about PFC, see the "Configuring Priority Flow Control" section.

Prerequisites for Network QoS

The network QoS policy has the following prerequisites:

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

Guidelines and Limitations

The network QoS policy has the following configuration guidelines and limitations:

- **show** commands with the **internal** keyword are not supported.
- Changing the network QoS policy is a disruptive operation, and it can cause traffic drops on any or all ports.

- When enabling jumbo MTU, the default network QoS policy can support jumbo frames. Under the network QoS policy, the MTU is used only for buffer carving when no-drop classes are configured. No additional MTU adjustments are required under the network QoS policy to support jumbo MTU.
- Network QoS is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3).

Configuring Network QoS Policies

You can configure a network QoS policy by following one of these methods:

- Predefined policies—You can apply a predefined network QoS policy that fits your requirement. By default, default-nq-policy is configured.
- User-defined policy—You can create a network QoS policy that conforms to one of the system-defined policies.

Copying a Predefined Network QoS Policy

SUMMARY STEPS

1. **qos copy policy-map type network-qos default-nq-policy** {**prefix** *prefix* | **suffix** *suffix*}
2. **show policy-map type network-qos my_nq**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**qos copy policy-map type network-qos default-nq-policy** {**prefix** *prefix* | **suffix** *suffix*} **Example:** `switch# qos copy policy-map type network-qos default-nq-policy prefix my_nq`	Copies a predefined network QoS policy and adds a suffix or prefix to its name. A prefix or suffix name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
Step 2	**show policy-map type network-qos my_nq** **Example:** `switch# show policy-map type network-qos my_nq`	(Optional) Displays the type network-qos policy map.

Configuring a User-Defined Network QoS Policy

SUMMARY STEPS

1. **configure terminal**
2. **class-map type network-qos match-any** *class-name*
3. **match qos-group** *group*
4. **exit**
5. **policy-map type network-qos** *policy-map-name*

6. **class type network-qos** {*class-name* | **class-default**}
7. **pause** *group*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map type network-qos match-any** *class-name* **Example:** `switch(config)# class-map type network-qos match-any c-nq2` `switch(config-cmap-nqos)#`	Configures the class map of the type network-qos and enters class-map mode. Class network-qos names are listed in previous System-Defined Type network-qos Class Maps table.
Step 3	**match qos-group** *group* **Example:** `switch(config-cmap-nqos)# match qos-group 2`	Specifies the QoS group to match. The range is from 0 to 3.
Step 4	**exit** **Example:** `switch (config-cmap-nqos)# exit` `switch (config)#`	Exits class-map mode and enters global configuration mode.
Step 5	**policy-map type network-qos** *policy-map-name* **Example:** `switch(config)# policy-map type network-qos map2`	Creates a policy map. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
Step 6	**class type network-qos** {*class-name*	**class-default**} **Example:** `switch(config-pmap-nqos)# class type network-qos c1-nq2`	Refers to the class map of type network-qos as configured in Step 2.
Step 7	**pause** *group* **Example:** `switch(config-pmap-nqos-c)# pause pfc-cos 2`	Specifies no-drop for the QoS group. **Note** For 7.0(3)I1(1) and earlier, the no-drop queuing configuration is not supported in the network-qos policy for the Cisco Nexus 9300 platform.

Applying a Network QoS Policy on a System

You apply a network QoS policy globally on a system. Applying a network QoS policy also automatically applies the corresponding queuing policies.

SUMMARY STEPS

1. **configure terminal**
2. **system qos**
3. **service-policy type network-qos** {*policy-map-name* | **default-nq-policy**}

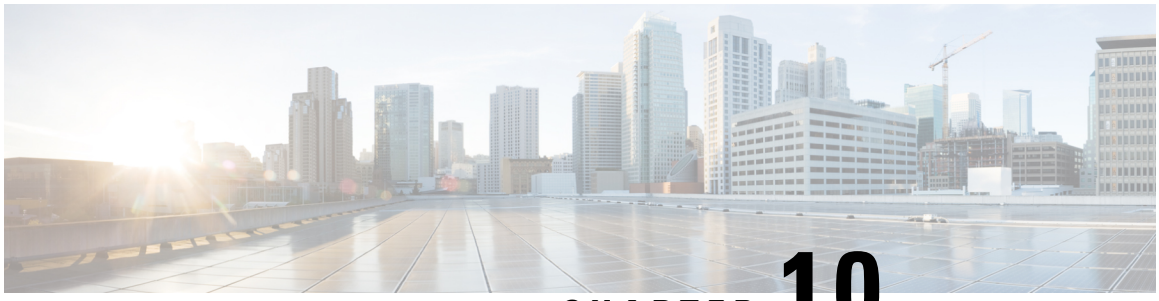
DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**system qos** **Example:** `switch (config)# system qos` `switch (config-sys-qos)#`	Enters system qos mode.
Step 3	**service-policy type network-qos** {*policy-map-name* | **default-nq-policy**} **Example:** `switch (config-sys-qos)# service-policy type network-qos map1`	Specifies the policy map to use as the service policy for the system. **Note** To restore the system to the default network QoS service policy, use the **no** form of this command. **Note** All Layer 4 class-maps under the network-qos policy-map must be configured before applying it under the system qos level.

Verifying the Network QoS

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

Command	Purpose
show class-map type network-qos	Displays the type network-qos class maps.
show policy-map type network-qos	Displays the type network-qos policy maps.
show policy-map system type network-qos	Displays the active type network-qos class maps.



CHAPTER 10

Configuring Link Level Flow Control

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 Restrictions for Link Level Flow Control

- **show** commands with the **internal** keyword are not supported.
- Link-level flow control (LLFC) is supported on Cisco Nexus 9500 platform switches with Network Forwarding Engine (NFE) (and Cisco Nexus 3164Q switch with NFE) (6.1(2)I3(4) and later releases.
- Changing or configuring LLFC on FEX HIF or FEX HIF PO interfaces is not supported.
- Beginning with Cisco NX-OS Release 7.0(3)I5(1) and later releases, Link-level flow control (LLFC) is supported on the Cisco Nexus 9300 and 9300-EX platform switches.
- Ethernet interfaces do not auto-detect the link-level flow control capability. You must configure the capability explicitly.
- Enabling link level flow control requires a part of the buffer to be reserved. This reduces the available shared buffer space.
- Data Center Bridging Exchange Protocol (DCBX) is not supported.
- Configuration time quanta of the pause frames is not supported.

- On each Ethernet interface, the switch can enable either PFC or LLFC, but not both.

Note When both PFC and LLFC are enabled, LLFC is selected.

- Only pure CoS-based classification of traffic classes is 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.
- Weighted Random Early Detection (WRED) should not be enabled on a no-drop class because it can cause egress queue drops.
- 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.
- 3232C does not support a combination of cut-through and LLFC enabled ports. Cut-through and LLFC are mutually exclusive and will work without the presence of the other feature. Post 9.3(8) release, on a cut-through enabled switch, if LLFC is enabled on a port, that port will operate in store and forward mode.
- First generation Cisco Nexus 9000 switches (NFE-based) can properly operate in cut-through switching mode only when there is no back pressure, i.e. LLFC pause frames should not be coming in. If RX LLFC is enabled alongside the default switching mode of cut-through, in some circumstances this can lead to output errors on the port configured with RX LLFC. To mitigate that, the user needs to either configure store and forward switching mode via the CLI "**switching-mode store-forward**" or disable RX LLFC on the given port. This limitation applies to the following PIDs:
  - N9K-C93120TX, N9K-C93128TX, N9K-C9332PQ, N9K-C9372PX, N9K-C9372PX-E, N9K-C9372TX, N9K-C9372TX-E, N9K-C9396PX, N9K-C9396TX, N9K-X9408PC-CFP2, N9K-X9432PQ, N9K-X9464PX, N9K-X9464TX, N9K-X9464TX2, N9K-X9536PQ, N9K-X9564PX, N9K-X9564TX, and N9K-X9636PQ.

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.

Switch A	Switch B	Description
LLFC configured to receive and transmit PAUSE frames.	LLFC configured to receive PAUSE frames.	Switch A can transmit 802.3x PAUSE frames and honor 802.3x PAUSE frames. Switch B can only receive 802.3x PAUSE frames.
LLFC configured to receive and transmit PAUSE frames.	LLFC configured to transmit PAUSE frames.	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.

How to Configure Link Level Flow Control

Configuring Link Level Flow Control Receive

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface ethernet 1/1** **Example:** `Device(config)# interface ethernet 1/1`	Configures an interface type and enters interface configuration mode.
Step 3	**flowcontrol receive on** **Example:** `Device(config-if)# flowcontrol receive on`	Enables the interface to receive and process pause frames.
Step 4	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode.

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. **configure terminal**
2. **interface ethernet 1/1**
3. **flowcontrol send on**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**interface ethernet 1/1** **Example:** `Device(config)# interface ethernet 1/1`	Configures an interface type and enters interface configuration mode.
Step 3	**flowcontrol send on** **Example:** `Device(config-if)# flowcontrol transmit on`	Enables the interface to send pause frames to remote devices.
Step 4	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.

Configuration Examples for Link Level Flow Control

Example: Configuring a No-Drop Policy

Configuring a No-Drop Policy

The following example shows how to configure a no-drop policy and attach the policy to a session policy:

```
Device# configure terminal
Device(config)# class-map type network-qos class1
Device(config-cmap-nq)# match qos-group 1
Device(config-cmap-nq)# policy-map type network-qos my_network_policy
Device(config-pmap-nq)# class type network-qos class1
Device(config-pmap-nq-c)# pause pfc-cos 2
Device(config-pmap-nq-c)# system qos
Device(config-sys-qos)# service-policy type network-qos my_network_policy
Device# show running ipqos
```

Classifying Traffic to a No-Drop Class

The following example shows how to create a QoS policy to map all the traffic to the no-drop class:

```
Device# configure terminal
Device(config)# class-map type qos class1
Device(config-cmap-qos)# match cos 2
Device(config-cmap-qos)# policy-map type qos my_qos_policy
Device(config-pmap-qos)# class type qos class1
Device(config-pmap-c-qos)# set qos-group 1
```

```
Device(config-pmap-c-qos)# interface e1/5
Device(config-sys-qos)# service-policy type qos input my_qos_policy
Device(config-sys-qos)#
```

Add the queuing policy that guarantees the bandwidth for qos-group 1 and apply that under system-qos as outlined in the following example:

```
policy-map type queuing my_queuing_policy
class type queuing c-out-q-default
bandwidth percent 1
class type queuing c-out-q3
bandwidth percent 0
class type queuing c-out-q2
bandwidth percent 0
class type queuing c-out-q1
bandwidth percent 99

system qos
  service-policy type queuing output my_queuing_policy
```

In the above example, c-out-q1 by default matches the traffic on qos-group 1. Therefore, the non-default class-map for queuing which matches qos-group 1 is not needed. For further information on configuring queuing, see Configuring Queuing.

For LLFC to be enabled, you need to configure the no-drop policy on network-qos. The buffering module needs to inform the MAC module to generate pause (either LLFC or PFC based on the interface level configuration). PFC negotiation to the adapter is by using DCBX. LLFC or PFC is controlled by the configuration on the interfaces. For example, the **flow-control send and receive on** enables LLFC on the interfaces and the **priority-flow-control mode on** enables PFC on the interfaces.

If DCBX is supported, auto mode negotiates the PFC with the adapter. This is the interface level configuration to enable LLFC or PFC but regardless of it, you have to configure network-qos level pause configuration for LLFC to work. Even if the traffic is classified to qos-group 1 but when it generates pause, it generates LLFC based on the interface level configuration.

Example: Configuring Link Level Flow Control Receive and Send

Configuring Link Level Flow Control Receive and Send

The following examples show how to configure Link Level Flow Control receive and send on the device.

- When only LLFC receive is enabled, no-drop class does not need to be configured on the system network-qos.

Note You must configure the no-drop class under system network-qos on the Cisco Nexus 9200 and 9300-EX/FX platforms for releases earlier than NX-OS 7.0(3)I7(3).

```
Device# configure terminal
Device(config)# interface ethernet 1/1
```

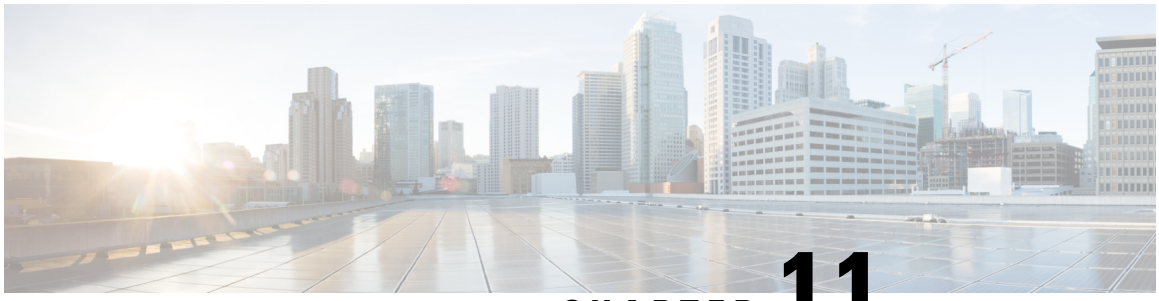
```
Device(config-if)# flowcontrol receive on
Device(config-if)# exit
```

- When both LLFC receive and send are enabled, no-drop class needs to be configured on the system network-qos. (Refer to the Configuring a No-Drop Policy example for information about configuring the no-drop class.)

```
Device# configure terminal
Device(config)# interface ethernet 1/1
Device(config-if)# flowcontrol receive on
Device(config-if)# flowcontrol send on
Device(config-if)# exit
```

- When only LLFC send is enabled, no-drop class needs to be configured on the system network-qos. (Refer to the Configuring a No-Drop Policy example for information about configuring the no-drop class.)

```
Device# configure terminal
Device(config)# interface ethernet 1/1
Device(config-if)# flowcontrol send on
Device(config-if)# exit
```



CHAPTER 11

Configuring Priority Flow Control

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.

Note Cisco Nexus 9000 Series switches support the transport of RDMA over Converged Ethernet (RoCE) v1 and v2 protocols.

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 has the following configuration guidelines and limitations:

- PFC is not supported on the Cisco Nexus 9508 switch (NX-OS 7.0(3)F3(3).
- The **show** commands with the **internal** keyword are not supported.
- Adding pause buffer size threshold configuration is optional for cable lengths that are less than 100 meters and it need not be configured.
- Input queuing policy maps cannot have pause buffer and priority/bandwidth together.
- For cable lengths that are greater than 100m, the pause buffer size threshold configuration is mandatory and it is required as part of the 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.
- You can configure a PFC watchdog interval to detect whether packets in a no-drop queue are being drained within a specified time period. When the time period is exceeded, all outgoing packets are dropped on interfaces that match the PFC queue that is not being drained. Beginning with Cisco NX-OS Release 7.0(3)I4(2), this feature is supported only for Cisco Nexus 9200 Series switches, Cisco Nexus 93108TC-EX, and 93180YC-EX switches, and Cisco Nexus 9508 switches with the X9732C-EX line cards.
- 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 a 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 three 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
  - Number of 10G and 40G ports
- 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 input queuing 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.
- Dynamic load balancing cannot be enabled for internal links with PFC. You must disable DLB and enable RTAG7 load-balancing for internal links with the port-channel load-balance internal rtag7 command.
- The dynamic load balancing (DLB) based hashing scheme is enabled by default on all internal links of a linecard. When DLB is enabled, no-drop traffic might experience out-of-order packet delivery when congestion on internal links occurs and PFC is applied. If applications on the system are sensitive to out-of-order delivery, you can adjust for this by disabling DLB at the qos-group level. Disable DLB by using the **set dlb-disable** action in the QoS policy-maps and the **set qos-group** action for no-drop classes.

  In the following example assume that qos-group 1 is a no-drop class. DLB is disabled for this no-drop class by adding the **set dlb-disable** action and the **set qos-group** action.

```
switch(config)# policy-map p1
switch(config-pmap-qos)# class c1
switch(config-pmap-c-qos)# set qos-group 1
switch(config-pmap-c-qos)# set dlb-disable
switch(config-pmap-c-qos)# end
switch# show policy-map p1


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

```
policy-map type qos p1
  class  c1
    set qos-group 1
    set dlb-disable
```

Note The following Cisco Nexus platform switches do not support the **set-dlb-disable** command:

- Cisco Nexus 9200-series platform switches
- Cisco Nexus 9300-EX/FX/FX2 platform switches
- Cisco Nexus 9500-series platform switches with -EX and -FX line cards

- 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 36: Default PFC Setting

Parameter	Default
PFC	Auto

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

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**interface** *type slot/port* **Example:** `switch(config)# interface ethernet 2/5` `switch(config-if)#`	Enters interface mode on the interface specified.
Step 3	**priority-flow-control mode** [| **off** |**on**] **Example:** `switch(config-if)# priority-flow-control mode on` `switch(config-if)#`	Sets PFC to the on mode.
Step 4	**show interface priority-flow-control** **Example:** `switch# show interface priority-flow-control`	(Optional) Displays the status of PFC on all interfaces.

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 match** { **all** | **any** } *class-name*
3. **match cos** *cos-value*
4. **match dscp** *dscp-value*
5. **exit**
6. **policy-map type qos** *policy-name*
7. **class** *class-name*
8. **set qos-group** *qos-group-value*
9. **exit**
10. **exit**
11. **policy-map type network-qos** *policy-name*
12. **class type network-qos** *class-name*
13. **pause pfc-cos** *value* [**receive**]
14. **exit**
15. **exit**
16. **system qos**
17. **service-policy type network-qos** *policy-name*

18. **exit**
19. **interface ethernet** *slot* / *number*
20. **priority-flow-control mode** { **auto** | **on** | **off** }
21. **service-policy type qos input** *policy-name*
22. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	**class-map type qos match** { **all** | **any** } *class-name* **Example:** `switch(config)# class-map type qos c1` `switch(config-cmap-qos)#`	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. **match** { **all** | **any** }: Default is **match all** (if multiple matching statements are present all of them must be matched).
Step 3	**match cos** *cos-value* **Example:** `switch(config-cmap-qos)# match cos 2` `switch(config-cmap-qos)#`	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.
Step 4	**match dscp** *dscp-value* **Example:** `switch(config-cmap-qos)# match dscp 3` `switch(config-cmap-qos)#`	Specifies the DSCP value to match for classifying packets into this class. You can configure a DSCP value in the range of 0 to 63 or the listed values.
Step 5	**exit** **Example:** `switch(config-cmap-qos)# exit` `switch(config)#`	Exits class-map mode and enters global configuration mode.
Step 6	**policy-map type qos** *policy-name* **Example:** `switch(config)# policy-map type qos p1` `switch(config-pmap-qos)#`	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.
Step 7	**class** *class-name* **Example:** `switch(config-pmap-qos)# class c1` `switch(config-pmap-c-qos)#`	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.

	Command or Action	Purpose
Step 8	**set qos-group** *qos-group-value* **Example:** `switch(config-pmap-c-qos)# set qos-group 3` `switch(config-pmap-c-qos)#`	Configures one or more qos-group values to match on for classification of traffic into this class map. There is no default value.
Step 9	**exit** **Example:** `switch(config-pmap-c-qos)# exit` `switch(config-pmap-qos)#`	Exits the system class configuration mode and enters policy-map mode.
Step 10	**exit** **Example:** `switch(config-pmap-qos)# exit` `switch(config)#`	Exits policy-map mode and enters global configuration mode.
Step 11	**policy-map type network-qos** *policy-name* **Example:** `switch(config)# policy-map type network-qos pfc-qos` `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, are case sensitive, and can be up to 40 characters.
Step 12	**class type network-qos** *class-name* **Example:** `switch(config-pmap-nqos)# class type network-qos nw-qos3` `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 13	**pause pfc-cos** *value* [**receive**] **Example:** `switch(config-pmap-nqos-c)# pause pfc-cos 3 receive` `switch(config-pmap-nqos-c)#`	PFC sends a pause frame that indicates which CoS value needs to be paused. Only PFC receive is enabled for the list of PCF CoS values. **receive**: When this optional keyword is used, PFC only receives and honors pause frames. PFC will never send pause frames. This is known as "Asymmetric PFC". **Note** Although not required, the **pause pfc-cos** *value* should match the *qos-group-value* in the **set qos-group** command. See the **set qos-group** command in steps 8 above.
Step 14	**exit** **Example:** `switch(config-pmap-nqos-c)# exit` `switch(config-pmap-nqos)#`	Exits configuration mode and enters policy-map mode.
Step 15	**exit** **Example:** `switch(config-pmap-nqos)# exit` `switch(config)#`	Exits policy-map mode and enters global configuration mode.

	Command or Action	Purpose
Step 16	**system qos** **Example:** `switch(config)# system qos` `switch(config-sys-qos)#`	Enters system class configuration mode.
Step 17	**service-policy type network-qos** *policy-name* **Example:** `switch(config-sys-qos)# service-policy type network-qos pfc-qos`	Applies the policy map of type network-qos at the system level or to the specific interface.
Step 18	**exit** **Example:** `switch(config-sys-qos)# exit` `switch(config)#`	Exits policy-map mode and enters global configuration mode.
Step 19	**interface ethernet** *slot* / *number* **Example:** `switch(config)# interface ethernet 1/1` `switch(config-if)#`	Enters the ethernet interface configuration mode for the selected slot and chassis number.
Step 20	**priority-flow-control mode** { **auto** | **on** | **off** } **Example:** `switch(config-if)# priority-flow-control mode on` `switch(config-if)#`	Enables the priority flow control policy for the interface.
Step 21	**service-policy type qos input** *policy-name* **Example:** `switch(config-if)# service-policy type qos input p1`	Adds classification to the interface ensuring that packets matching the previously configured CoS or DSCP values are classified in the correct QoS group.
Step 22	**exit** **Example:** `switch(config-if)# exit` `switch(config)#`	Exits the ethernet interface mode and enters the global configuration mode.

Configuring Pause Buffer Thresholds and Queue Limit Using Ingress Queuing Policy

The pause buffer thresholds specified in the network-qos policy are shared by all the ports in the system. However, there are situations where a few ports may need different thresholds (such as long distance connections). An ingress queuing policy can be used for this purpose.

An ingress queuing policy also allows the configuration of the queue-limit to restrict the amount of shared buffer that can be used in addition to the reserved pause buffer by the no-drop class.

Each no-drop class is mapped internally to one of the port's priority-group in the ingress direction. The configured pause buffer thresholds and queue-limit are applied to the priority-group associated with the class.

Note 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 QoS policy configuration.

Note About queue limits for 100G enabled devices (such as the Cisco Nexus 9300 platform switch with the N9K-M4PC-CFP2 GEM):

- The maximum dynamic queue-limit alpha value supported by the device might be greater that 8. However 8 is the maximum alpha value supported. Configuring the alpha value to a value greater than 8 is overridden by the maximum alpha value of 8.

  No message is issued when the alpha value is overridden.

- The static queue-limit has a maximum of 20,000 cells. Any value specified greater than the maximum 20,000 cell limit is overridden by the 20,000 cell limit.

  No message is issued when the cell limit is overridden.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map type queuing** *policy-map-name*
3. **class type queuing** *c-in-q1*
4. **pause buffer-size** *buffer-size* **pause threshold** *xoff-size* **resume threshold** *xon-size*
5. **no pause buffer-size** *buffer-size* **pause threshold** *xoff-size* **resume threshold** *xon-size*
6. **queue-limit** *queue size* [**dynamic** *dynamic threshold*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**policy-map type queuing** *policy-map-name*	Enters policy-map queuing class mode and identifies the policy map assigned to the type queuing policy map.
Step 3	**class type queuing** *c-in-q1*	Attaches the class map of type queuing and then enters policy-map class queuing mode. Class queuing names are listed in the System-Defined Type queuing Class Maps table. **Note** The qos-group associated with the class must be defined as a no-drop class in the network-qos policy applied in the system qos.

	Command or Action	Purpose
Step 4	**pause buffer-size** *buffer-size* **pause threshold** *xoff-size* **resume threshold** *xon-size*	Specifies the buffer threshold settings for pause and resume.
Step 5	**no pause buffer-size** *buffer-size* **pause threshold** *xoff-size* **resume threshold** *xon-size*	Removes the buffer threshold settings for pause and resume.
Step 6	**queue-limit** *queue size* [**dynamic** *dynamic threshold*]	(Optional) Specifies either the static or dynamic shared limit available to the ingress priority-group. The static queue limit defines the fixed size to which the priority-group can grow. The dynamic queue limit allows the priority-group's threshold size to be decided depending on the number of free cells available, in terms of the alpha value. **Note** Cisco Nexus 9200 platform switches only support a class level dynamic threshold configuration with respect to the alpha value. This means that all ports in a class share the same alpha value.

Verifying the Priority Flow Control Configuration

To display the PFC configuration, perform the following task:

Command	Purpose
show interface priority-flow-control [**module** *number*]	Displays the status of PFC on all interfaces or on specific modules.

Configuration Examples for Priority Flow Control

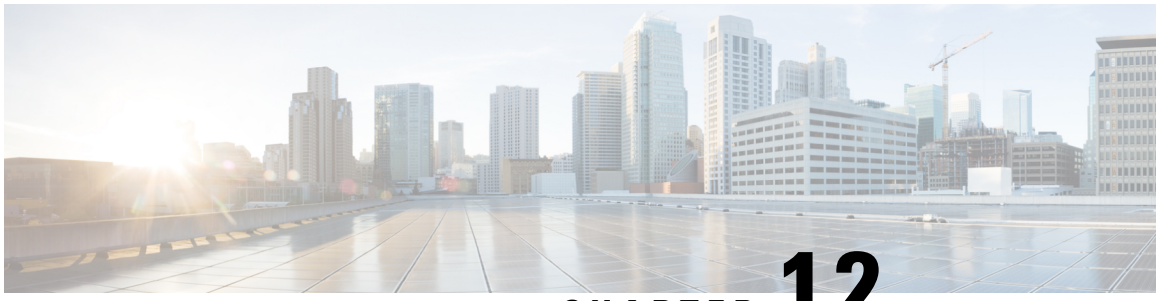
The following example shows how to configure PFC:

```
configure terminal
interface ethernet 5/5
priority-flow-control mode on
```

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

```
switch(config)# class-map type qos c1
switch(config-cmap-qos)# match cos 3
switch(config-cmap-qos)# exit
switch(config)# policy-map type qos p1
switch(config-pmap-qos)# class type qos c1
switch(config-pmap-c-qos)# set qos-group 3
switch(config-pmap-c-qos)# exit
switch(config-pmap-qos)# exit
switch(config)# class-map type network-qos match-any c1
switch(config-cmap-nqos)# match qos-group 3
switch(config-cmap-nqos)# exit
switch(config)# policy-map type network-qos p1
switch(config-pmap-nqos)# class type network-qos c-nq1
```

```
switch(config-pmap-nqos-c)# pause pfc-cos 3
switch(config-pmap-nqos-c)# exit
switch(config-pmap-nqos)# exit
switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos p1
```



CHAPTER 12

Monitoring QoS Statistics

About QoS Statistics

You can display various QoS statistics for the device. By default, statistics are enabled, but you can disable this feature. For more information, see the Configuration Examples For Monitoring QoS Statistics section.

Prerequisites for Monitoring QoS Statistics

Monitoring QoS statistics has the following prerequisites:

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

Guidelines and Limitations

- **show** commands with the **internal** keyword are not supported.
- The **show queuing interface** command can display information about internal interfaces.

  The command format for this information is specified as **ii** *x*/*y*/*z*. Where *x* is the module number, *y* is the value 1, and *z* is the internal interface number within the module.

Note The number of internal interfaces within a module varies based on the type of the linecard.

Note Alternatively, you can display information about internal interfaces by providing the module number in the **show queuing** command so that queuing information for both front-panel and internal interfaces of a given module are displayed together.

Example:

```
switch# show queuing interface ii 4/1/2

slot  4
=======


Egress Queuing for ii4/1/2 [System]
------------------------------------------------------------------------
QoS-Group# Bandwidth% PrioLevel                Shape
                                    Min          Max         Units
------------------------------------------------------------------------
      3             -         1           -            -     -
      2             0         -           -            -     -
      1             0         -           -            -     -
      0           100         -           -            -     -
+-------------------------------------------------------------------+
|                             QOS GROUP 0                           |
+-------------------------------------------------------------------+
|                |  Unicast       | OOBFC Unicast  |  Multicast     |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|               0|          235775|
|        Tx Byts |               0|               0|        22634400|
|   Dropped Pkts |               0|               0|               0|
|   Dropped Byts |               0|               0|               0|
|   Q Depth Byts |               0|               0|               0|
+-------------------------------------------------------------------+
|                             QOS GROUP 1                           |
+-------------------------------------------------------------------+
|                |  Unicast       | OOBFC Unicast  |  Multicast     |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|               0|               0|
|        Tx Byts |               0|               0|               0|
|   Dropped Pkts |               0|               0|               0|
|   Dropped Byts |               0|               0|               0|
|   Q Depth Byts |               0|               0|               0|
+-------------------------------------------------------------------+
|                             QOS GROUP 2                           |
+-------------------------------------------------------------------+
|                |  Unicast       | OOBFC Unicast  |  Multicast     |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|               0|               0|
|        Tx Byts |               0|               0|               0|
|   Dropped Pkts |               0|               0|               0|
|   Dropped Byts |               0|               0|               0|
|   Q Depth Byts |               0|               0|               0|
+-------------------------------------------------------------------+
|                             QOS GROUP 3                           |
+-------------------------------------------------------------------+
|                |  Unicast       | OOBFC Unicast  |  Multicast     |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|               0|               0|
|        Tx Byts |               0|               0|               0|
```

```
|   Dropped Pkts |               0|               0|               0|
|   Dropped Byts |               0|               0|               0|
|   Q Depth Byts |               0|               0|               0|
+-------------------------------------------------------------------+
|                      CONTROL QOS GROUP                            |
+-------------------------------------------------------------------+
|                |  Unicast       | OOBFC Unicast  |  Multicast     |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|               0|               0|
|        Tx Byts |               0|               0|               0|
|   Dropped Pkts |               0|               0|               0|
|   Dropped Byts |               0|               0|               0|
|   Q Depth Byts |               0|               0|               0|
+-------------------------------------------------------------------+
|                         SPAN QOS GROUP                            |
+-------------------------------------------------------------------+
|                |  Unicast       | OOBFC Unicast  |  Multicast     |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|               0|               0|
|        Tx Byts |               0|               0|               0|
|   Dropped Pkts |               0|               0|               0|
|   Dropped Byts |               0|               0|               0|
|   Q Depth Byts |               0|               0|               0|
+-------------------------------------------------------------------+
Cannot get ingress statistics for if_index: 0x4a180001 Error 0xe

Port Egress Statistics
------------------------------------------------------
WRED Drop Pkts                              0

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

Enabling Statistics

You can enable or disable QoS statistics for all interfaces on the device. By default, QoS statistics are enabled.

SUMMARY STEPS

1. **configure terminal**
2. Enable or disable QoS statistics:
   - Enable QoS statistics:

     qos statistics

   - Disable QoS statistics:

no qos statistics

3. **show policy-map interface**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal` `switch(config)#`	Enters global configuration mode.
Step 2	Enable or disable QoS statistics: • Enable QoS statistics: **qos statistics** • Disable QoS statistics: **no qos statistics** **Example:** • Enable QoS statistics: `switch(config)# qos statistics` • Disable QoS statistics: `switch(config)# no qos statistics`	• Enable QoS statistics: Enables QoS statistics on all interfaces. • Disable QoS statistics: Disables QoS statistics on all interfaces.
Step 3	**show policy-map interface** **Example:** `switch(config)# show policy-map interface`	(Optional) Displays the statistics status and the configured policy maps on all interfaces.
Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the running configuration to the startup configuration.

Monitoring the Statistics

You can display QoS statistics for all interfaces or a selected interface, data direction, or a QoS type.

SUMMARY STEPS

1. **show policy-map** [*policy-map-name*] [**interface** [**input** | **output**]] [**type** {**control-plane** | **network-qos** | **qos** | **queuing**}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show policy-map** [*policy-map-name*] [**interface** [**input** | **output**]] [**type** {**control-plane** | **network-qos** | **qos** | **queuing**}] **Example:** `switch# show policy-map interface ethernet 2/1`	Displays statistics and the configured policy maps on all interfaces, the specified interface, or on a specified data direction or QoS type.

Clearing Statistics

You can clear QoS statistics for all interfaces or a selected interface, data direction, or QoS type.

SUMMARY STEPS

1. **clear qos statistics** [**interface** [**input** | **output**] [**type** {**qos** | **queuing**}]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**clear qos statistics** [**interface** [**input** | **output**] [**type** {**qos** | **queuing**}]] **Example:** `switch# clear qos statistics type qos`	Clears statistics and the configured policy maps on all interfaces or the specified interface or on a specified data direction or QoS type.

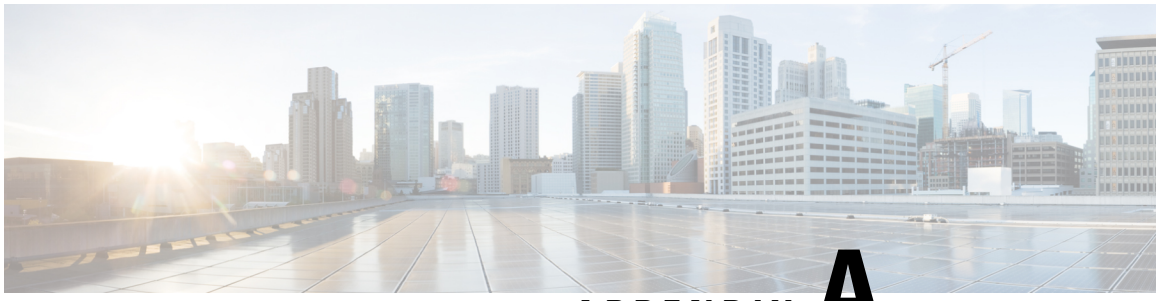
Configuration Examples For Monitoring QoS Statistics

The following example shows how to display the QoS statistics:

```
Global statistics status :   enabled

Ethernet6/1
          Service-policy (queuing) output:   default-out-policy

            Class-map (queuing):   c-out-q3 (match-any)
              priority level 1

            Class-map (queuing):   c-out-q2 (match-any)
              bandwidth remaining percent 0

            Class-map (queuing):   c-out-q1 (match-any)
              bandwidth remaining percent 0

            Class-map (queuing):   c-out-q-default (match-any)
              bandwidth remaining percent 100
```

The following example shows how to obtain information about queuing and PFC related counters:

```
switch(config-vlan-config)# show queuing interface ethernet 2/1

Egress Queuing for Ethernet2/1 [System]
------------------------------------------------------------------------
QoS-Group# Bandwidth% PrioLevel                Shape
                                     Min          Max        Units
------------------------------------------------------------------------
      3             -         1           -            -     -
      2             0         -           -            -     -
      1             0         -           -            -     -
      0           100         -           -            -     -
+-------------------------------------------------------------------+
|                              QOS GROUP 0                          |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|   Dropped Pkts |               0|
+-------------------------------------------------------------------+
|                              QOS GROUP 1                          |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|   Dropped Pkts |               0|
+-------------------------------------------------------------------+
|                              QOS GROUP 2                          |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|   Dropped Pkts |               0|
+-------------------------------------------------------------------+
|                              QOS GROUP 3                          |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|   Dropped Pkts |               0|
+-------------------------------------------------------------------+
|                      CONTROL QOS GROUP 4                          |
+-------------------------------------------------------------------+
|        Tx Pkts |              58|   Dropped Pkts |               0|
+-------------------------------------------------------------------+
|                         SPAN QOS GROUP 5                          |
+-------------------------------------------------------------------+
|        Tx Pkts |               0|   Dropped Pkts |             948|
+-------------------------------------------------------------------+
```



APPENDIX A

Additional References

This appendix contains additional information related to implementing QoS on the Cisco NX-OS device.

This appendix includes the following sections:

RFCs

RFCs	Title
RFC 2474	*Differentiated Services Field*
RFC 2475	*Architecture for Differentiated Services*
RFC 2697	*A Single Rate Three Color Marker*
RFC 2698	*A Dual Rate Three Color Marker*
RFC 3289	*Management Information Base for the Differentiated Services Architecture*

Additional References