Configuring QoS

This chapter describes how to configure quality of service (QoS) on Cisco Nexus 5000 Series switches. It contains the following sections:

Information About QoS

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

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

Note Standard Ethernet is a best-effort medium which means that it lacks any form of flow control. In the event of congestion or collisions, Ethernet will drop packets. The higher level protocols detect the missing data and retransmit the dropped packets.

Fibre Channel requires a reliable transport system that guarantees the delivery of every packet. To properly support FCoE, Ethernet has been enhanced with a priority flow control (PFC) mechanism to prevent congestion.

MQC

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

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

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

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

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

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

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

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

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

- network-qos—Defines MQC objects that you can use for system level related actions.

- qos—Defines MQC objects that you can use for classification.

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

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

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

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

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

System Classes

The system qos is a type of MQC target. You use a service-policy to associate a policy map with the system qos target. A system qos policy applies to all interfaces on the switch unless a specific interface has an overriding service-policy configuration. The system qos policies are used to define system classes, the classes of traffic across the entire switch, and their attributes. To ensure QoS consistency (and for ease of configuration),

the switch distributes the system class parameter values to all its attached network adapters using the Data Center Bridging Exchange (DCBX) protocol.

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

On the Cisco Nexus 5000 Series switch, a system class is uniquely identified by a qos-group value. A total of six system classes are supported. Two of the six system classes are defaults and are always present on the switch. Up to four additional system classes can be created by the administrator.

Default System Classes

The Cisco Nexus 5000 Series switch provides the following system classes:

- Drop system class

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

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

- FCoE system class (For the Cisco Nexus 5010 switch and the Cisco Nexus 5020 switch)

  All Fibre Channel and FCoE control and data traffic is automatically classified into the FCoE system class, which provides no-drop service.

  This class is created automatically when the system starts up (the class is named class-fcoe in the CLI).

  You cannot delete class-fcoe and you can only modify the IEEE 802.1p CoS value to associate with this class. This class is identified by qos-group 1.

  The switch classifies packets into the FCoE system class as follows:

  ◦ FCoE traffic is classified based on EtherType.

  ◦ Native Fibre Channel traffic is classified based on the physical interface type.

  > **Note**   The optional N5K-M1404 or N5K-M1008 expansion modules provide native ½/4-Gigabit Fibre Channel ports.

- FCoE system class (For the Cisco Nexus 5548 switch)

  For the Cisco Nexus 5548 switch, the class-fcoe is not automatically created. Before you enable FCoE on the Cisco Nexus 5548 switch running Cisco NX-OS Release 5.0(2)N1(1), you must enable class-fcoe in the three types of qos policies:

  ◦ type qos policy maps

  ◦ type network-qos policy map (attached to system qos)

  ◦ type queuing policy map (class-fcoe must be configured with a non-zero bandwidth percentage for input queuing policy maps.

    When class-fcoe is not included in the qos policies, vFC interfaces do not come up and increased drops occur.

Note The Cisco Nexus 5548 switch supports five user-defined classes and one default drop system class.

Policy Types

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

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

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

  - Classification—The traffic that matches this class are as follows:

    - QoS Group—A class-map of type network-qos identifies a system-class and is matched by its associated qos-group.

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

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

    - MTU—The MTU that needs to be enforced for the traffic that is mapped to a system class. Each system class has a default MTU and the system class MTU is configurable.

    - Multicast optimization—This configuration specifies if the performance of multicast traffic mapped to this class will be optimized.

    - Pause no-drop—No drop specifies lossless service for the system class. Drop specifies that tail drop is used (arriving packets are dropped when the queue reaches its allocated size) when a queue for this system class is full.

      An additional parameter pfc-cos can be configured. This parameter identifies the class of service (CoS) values to assert priority flow control (PFC) when traffic for a no-drop system class is not mapped based purely on CoS experiences congestion.

    - Beginning in Cisco NX-OS Release 5.0(2)N1(1), you can change the buffer for the no-drop class.

    - Queue Limit—This configuration specifies the number of buffers that need to be reserved to the queues of this system class. This option is not configurable for no-drop system classes.

    - Set CoS value—This configuration is used to mark 802.1p values for all traffic mapped to this system class. For the Cisco Nexus 5020 switch and the Cisco Nexus 5010 switch, the marking value for a system class needs to be unique and cannot be used as a marking value for any other system class. The marking value does not need to be unique for the Cisco Nexus 5548 switch.

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

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

- ◦ Classification—The traffic that matches this class are as follows:

  - ◦ QoS Group—A class-map of type queuing identifies a system-class and is matched by its associated qos-group.

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

  > **Note** These policies can be attached to the system qos target or to any interface. The output queuing policy is used to configure output queues on the switch associated with system classes. The input queuing policy is used to configure scheduling for queues in the CNA. The input queuing policy parameters are signaled to the CNA over the DCBX protocol.

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

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

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

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

  - ◦ Classification—The traffic that matches this class are as follows:

    - ◦ Access Control Lists—Classifies traffic based on the criteria in existing ACLs.

    - ◦ Class of Service—Matches traffic based on the CoS field in the frame header.

    - ◦ DSCP—Classifies traffic based on the Differentiated Services Code Point (DSCP) value in the DiffServ field of the IP header.

    - ◦ IP Real Time Protocol—Classifies traffic on the port numbers used by real-time applications.

    - ◦ Precedence—Classifies traffic based on the precedence value in the type of service (ToS) field of the IP header.

    - ◦ Protocol—Classifies traffic based on the protocol field of the IP header.

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

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

- QoS Group—Sets the qos-group corresponding to the system class this traffic flow is mapped to.

  - Cisco Nexus 3000 Series supports the following:

    - 8 qos-groups

    - 8 queues for unicast

    - 4 queues for multicast

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

MTU

The Cisco Nexus 5000 Series switch is a Layer 2 switch, and it does not support packet fragmentation. A maximum transmission unit (MTU) configuration mismatch between ingress and egress interfaces may result in packets being truncated.

When configuring MTU, follow these guidelines:

- MTU is specified per system class. The system class allows a different MTU for each class of traffic but they must be consistent on all ports across the entire switch. You cannot configure MTU on the interfaces.

- Fibre Channel and FCoE payload MTU is 2158 bytes across the switch. As a result, the rxbufsize for Fibre Channel interfaces is fixed at 158 bytes. If the Cisco Nexus 5000 Series switch receives an rxbufsize from a peer that is different than 2158 bytes, it will fail the exchange of link parameters (ELP) negotiation and not bring the link up.

- Enter the **system jumbomtu** command to define the upper bound of any MTU in the system. The system jumbo MTU has a default value of 9216 bytes. The minimum MTU is 21 58 bytes and the maximum MTU is 9216 bytes.

- The system class MTU sets the MTU for all packets in the class. The system class MTU cannot be configured larger than the global jumbo MTU.

- The FCoE system class (for Fibre Channel and FCoE traffic) has a default MTU of 2158 bytes. This value cannot be modified.

- The default drop system class has a default MTU of 1500 bytes. You can configure this value.

- The switch sends the MTU configuration to network adapters that support DCBX.

> **Note** MTU is not supported in Converged Enhanced Ethernet (CEE) mode for DCBX.

Trust Boundaries

The trust boundary is enforced by the incoming interface 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. The 802.1p CoS and DSCP are preserved unless the marking is configured. There is no default CoS to queue and DSCP to queue mapping. You can define and apply a policy to create these mappings. By default, without a user defined policy, all traffic is assigned to the default queue.

- Any packet that is not tagged with an 802.1p CoS value is classified into the default drop system class. If the untagged packet is sent over a trunk, it is tagged with the default untagged CoS value, which is zero.

- You can override the default untagged CoS value for an Ethernet interface or port channel.

- You can override the default untagged CoS value for an Ethernet interface or a port channel interface using the **untagged cos** *cos-value* command.

- You can override the default untagged Cos value for an Ethernet or a Layer 3 interface or a port channel interface using the **untagged cos** *cos-value* command.

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

Ingress Queuing Policies

You can associate an ingress policy map with an Ethernet interface to guarantee bandwidth for the specified traffic class or to specify a priority queue.

The ingress policy is applied in the adapter to all outgoing traffic that matches the specified CoS value.

When you configure an ingress policy for an interface, the switch sends the configuration data to the adapter. If the adapter does not support the DCBX protocol or the ingress policy type-length-value (TLV), the ingress policy configuration is ignored.

Ingress Classification Policies

You use classification to partition traffic into classes. You classify the traffic based on the port characteristics (CoS field) or the packet header fields that include IP precedence, Differentiated Services Code Point (DSCP), and Layer 2 to Layer 4 parameters. The values used to classify traffic are called match criteria. When you define a traffic class, you can specify multiple match criteria, you can choose to not match on a particular criterion, or you can determine traffic class by matching any or all criteria.

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

Egress Queuing Policies

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

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

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

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

- FCoE traffic (traffic that maps to the FCoE system class) is assigned a queue. This queue uses weighted round-robin (WRR) scheduling with 50 percent of the bandwidth.

- Standard Ethernet traffic in the default drop system class is assigned a queue. This queue uses WRR scheduling with 50 percent of the bandwidth.

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

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

Policy for Fibre Channel Interfaces

The egress queues are not configurable for native Fibre Channel interfaces. Two queues are available as follows:

- A strict priority queue to serve high-priority control traffic.

- A queue to serve all data traffic and low-priority control traffic.

QoS for Traffic Directed to the CPU

The switch automatically applies QoS policies to traffic that is directed to the CPU to ensure that the CPU is not flooded with packets. Control traffic, such as BPDU frames, is given higher priority to ensure delivery.

QoS Configuration Guidelines and Limitations

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

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

-

- If less than four Ethernet classes are defined, up to two of these classes can be configured as no-drop classes. If more than three Ethernet classes are defined, only one of these classes can be configured as a no-drop class. The default drop class is counted as an Ethernet class.

- If priority flow control is enabled on an Ethernet interface, pause will never be applied to traffic with a drop system class. PFC does not apply pause to drop classes and the link-level pause feature is never enabled on an interface with PFC.

- All FCoE traffic on an Ethernet interface is mapped to one no-drop system class. By default, this class is associated with CoS value 3, although you can configure a different value. If you configure standard Ethernet traffic to use the same CoS value as FCoE, this traffic is still mapped to the FCoE system class and the switch will apply priority flow control on the FCoE CoS value.

- When a Cisco Nexus 2148T Fabric Extender is connected and in use, data traffic should never be marked with a CoS value of 7. CoS 7 is reserved for control traffic transiting the Fabric Extender.

- On the Cisco Nexus 5548 Switch, the FCoE class-fcoe system class is not enabled by default in the QoS configuration. Before enabling FCoE, you must include class-fcoe in each of the three policy types (network-qos, queuing, and qos).

Note   Type qos policies can be activated only on Cisco Nexus 5000 Series interfaces and Cisco Nexus 2000 Series Fabric Extender interfaces. Type qos policies on Fabric Extender fabric interfaces or Fabric Extender fabric EtherChannel interfaces are ineffective, though the Cisco NX-OS CLI does not reject the configuration.

We recommend that you do not configure type qos policy-maps on Fabric Extender fabric interfaces or Fabric Extender fabric EtherChannel interfaces to avoid wasting hardware resources.

When configuring EtherChannels, note the following guidelines:

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

- The priority flow control configured on an EtherChannel applies to all member interfaces.

Configuring System Classes

Configuring Class Maps

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

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

SUMMARY STEPS

1. switch# **configure terminal**

2. switch(config)# **class-map** [**type** {**network-qos** | **qos** | **queuing**}] *class-map name*

3. (Optional) switch(config)# **class-map** [**type qos**] [**match-all** | **match-any**] *class-map name*

4. (Optional)   switch(config)# **no class-map** [**type** {**network-qos** | **qos** | **queuing**}] *class-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **class-map** [**type** {**network-qos** | **qos** | **queuing**}] *class-map name*	Creates or accesses a named object that represents the specified class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters. The three class-map configuration modes are as follows: • network-qos—Network-wide (global) mode. CLI prompt: switch(config-cmap-nq)# • qos—Classification mode; this is the default mode. CLI prompt: switch(config-cmap-qos)# • queuing—Queuing mode. CLI prompt: switch(config-cmap-que)#
Step 3	switch(config)# **class-map** [**type qos**] [**match-all** | **match-any**] *class-map name*	(Optional) Specifies that packets must match any or all criteria that is defined for a class map. The default is match-all. • match-all—Classifies traffic if packets match all criteria that is defined for a specified class map.(for example, if the defined CoS and the ACL criteria matches) • match-any—Classifies traffic if packets match any criteria that is defined for a specified class map. (for example, if the CoS or the ACL criteria matches).
Step 4	switch(config)# **no class-map** [**type** {**network-qos** | **qos** | **queuing**}] *class-name*	(Optional) Deletes the specified class map. **Note** You cannot delete the two system-defined class maps: class-fcoe and class-default.

Configuring ACL Classification

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

SUMMARY STEPS

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

DETAILED STEPS

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

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

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

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

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

Configuring CoS Classification

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

Note The Cisco Nexus 2148 Fabric Extender does not support frames with the dot1p vlan 0 tag.

If a system class is configured with a no-drop function, the **match cos** command serves an additional purpose. The switch sends the CoS value to the adapter so that the adapter will apply a PFC pause for this CoS value.

The FCoE system class has a default CoS value of 3. You can add a **match cos** configuration to the FCoE system class to set a different CoS value. A PFC pause will be applied to traffic that matches the new value.

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **class-map type qos** *class-name*	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.
Step 3	switch(config-cmap-qos)# **match cos** *cos-value*	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.
		Note When a Cisco Nexus 2148T Fabric Extender is connected and in use, data traffic should never be marked with a CoS value of 7. CoS 7 is reserved for control traffic transiting the Fabric Extender.
Step 4	switch(config-cmap-qos)# **no match cos** *cos-value*	(Optional) Removes the match from the traffic class.

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

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

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

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

Configuring DSCP Classification

You can classify traffic based on the Differentiated Services Code Point (DSCP) value in the DiffServ field of the IP header (either IPv4 or IPv6). The following table shows the standard DSCP values:

Table 1: Standard DSCP Values

Value	List of DSCP Values
af11	AF11 dscp (001010)—decimal value 10
af12	AF12 dscp (001100)—decimal value 12
af13	AF13 dscp (001110)—decimal value 14
af21	AF21 dscp (010010)—decimal value 18
af22	AF22 dscp (010100)—decimal value 20
af23	AF23 dscp (010110)—decimal value 22
af31	AF31 dscp (011010)—decimal value 26
af32	AF40 dscp (011100)—decimal value 28

Value	List of DSCP Values
af33	AF33 dscp (011110)—decimal value 30
af41	AF41 dscp (100010)—decimal value 34
af42	AF42 dscp (100100)—decimal value 36
af43	AF43 dscp (100110)—decimal value 38
cs1	CS1 (precedence 1) dscp (001000)—decimal value 8
cs2	CS2 (precedence 2) dscp (010000)—decimal value 16
cs3	CS3 (precedence 3) dscp (011000)—decimal value 24
cs4	CS4 (precedence 4) dscp (100000)—decimal value 32
cs5	CS5 (precedence 5) dscp (101000)—decimal value 40
cs6	CS6 (precedence 6) dscp (110000)—decimal value 48
cs7	CS7 (precedence 7) dscp (111000)—decimal value 56
default	Default dscp (000000)—decimal value 0
ef	EF dscp (101110)—decimal value 46

SUMMARY STEPS

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

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	switch(config)# **class-map type qos** *class-name*	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.
Step 3	switch(config-cmap-qos)# **match dscp** *dscp-list*	Configures the traffic class by matching packets based on the values in the *dscp-list*. The standard DSCP values are shown in the preceding table.
Step 4	switch(config-cmap-qos)# **no match dscp** *dscp-list*	(Optional) Removes the match from the traffic class.

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

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

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

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

Configuring IP RTP Classification

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

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **class-map type qos** *class-name*	Creates a named object that represents a class of traffic. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.

	Command or Action	Purpose
Step 3	switch(config-cmap-qos)# **match ip rtp** *port-number*	Configures the traffic class by matching packets based on a range of lower and upper UDP port numbers, which is likely to target applications using RTP. Values can range from 2000 to 65535.
Step 4	switch(config-cmap-qos)# **no match ip rtp** *port-number*	(Optional) Removes the match from the traffic class.

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

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

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

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

Configuring Precedence Classification

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

Table 2: Precedence Values

Value	List of Precedence Values
<0-7>	IP precedence value
critical	Critical precedence (5)
flash	Flash precedence (3)
flash-override	Flash override precedence (4)
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. switch# **configure terminal**
2. switch(config)# **class-map type qos** *class-name*
3. switch(config-cmap-qos)# **match precedence** *precedence-values*
4. (Optional)   switch(config-cmap-qos)# **no match precedence** *precedence-values*

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **class-map type qos** *class-name*	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.
Step 3	switch(config-cmap-qos)# **match precedence** *precedence-values*	Configures the traffic class by matching packets based on *precedence-values* . Values are shown in the preceding table.
Step 4	switch(config-cmap-qos)# **no match precedence** *precedence-values*	(Optional) Removes the match from the traffic class.

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

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

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

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

Configuring Protocol Classification

You can classify traffic based on the protocol field in the IP header. The following table shows the protocol arguments:

Table 3: Protocol Arguments

Argument	Description
arp	Address Resolution Protocol (ARP)
clns_es	CLNS End Systems
clns_is	CLNS Intermediate System
dhcp	Dynamic Host Configuration (DHCP)

Argument	Description
ldp	Label Distribution Protocol (LDP)
netbios	NetBIOS Extended User Interface (NetBEUI)

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **class-map type qos** *class-name*
3. switch(config-cmap-qos)# **match protocol** {**arp** | **clns_es** | **clns_is** | **dhcp** | **ldp** | **netbios**}
4. (Optional) switch(config-cmap-qos)# **no match protocol** {**arp** | **clns_es** | **clns_is** | **dhcp** | **ldp** | **netbios**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **class-map type qos** *class-name*	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.
Step 3	switch(config-cmap-qos)# **match protocol** {**arp**	**clns_es**	**clns_is**	**dhcp**	**ldp**	**netbios**}	Configures the traffic class by matching packets based on the specified protocol.
Step 4	switch(config-cmap-qos)# **no match protocol** {**arp**	**clns_es**	**clns_is**	**dhcp**	**ldp**	**netbios**}	(Optional) Removes the match from the traffic class.

This example shows how to classify traffic by matching packets based on the protocol field:

```
switch# configure terminal
switch(config)# class-map type qos class_protocol
switch(config-cmap-qos)# match protocol arp
```

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

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

Configuring QoS Group Classification

You can classify traffic based on the value of the QoS group internal label, that represents a system class. You can set the value of the QoS group within a policy map using the **set qos-group** command.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **class-map type** {**network-qos** | **queuing**} *class-name*
3. switch(config-cmap-que)# **match qos-group** *qos-group-value*
4. (Optional)   switch(config-cmap-que)# **no match qos-group** *qos-group-value*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **class-map type** {**network-qos** | **queuing**} *class-name*	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.
Step 3	switch(config-cmap-que)# **match qos-group** *qos-group-value*	Configures the traffic class by matching packets based on a list of QoS group values. Values can range from 0 to 5. QoS group 0 is equivalent to class-default and QoS group 1 is equivalent to class-fcoe. **Note** qos-groups 0 and 1 are reserved for default classes and cannot be configured.
Step 4	switch(config-cmap-que)# **no match qos-group** *qos-group-value*	(Optional) Removes the match from the traffic class.

This example shows how to classify traffic based on the value of the QoS group:

```
switch# configure terminal
switch(config)# class-map type queuing class_qos_group
switch(config-cmap-que)# match qos-group 4
```

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

```
switch# show class-map class_qos_group
```

Configuring Policy Maps

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

The switch provides two default system classes: a no-drop class for lossless service (class-fcoe) and a drop class for best-effort service (class-default). You can define up to four additional system classes for Ethernet traffic.

Note The Cisco Nexus 5548 switch supports five user-defined classes and one default drop system class.

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

- network-qos: default-nq-policy

- Input qos: default-in-policy

- Input queuing: default-in-policy

- Output queuing: default-out-policy

Beginning with Cisco NX-OS Release 5.0(2)N1(1), for the Cisco Nexus 5548 switch, there are four pre-defined policy maps for FCoE:

- service-policy type qos input fcoe-default-in-policy

- service-policy type queuing input fcoe-default-in-policy

- service-policy type queuing output fcoe-default-out-policy

- service-policy type network-qos fcoe-default-nq-policy

Note Before you enable FCoE on the Cisco Nexus 5548 switch running Cisco NX-OS Release 5.0(2)N1(1), you must enable class-fcoe in the three types of qos policies:

- type qos policy maps

- type network-qos policy map (attached to system qos)

- type queuing policy map (class-fcoe must be configured with a non-zero bandwidth percentage for input queuing policy maps.

When class-fcoe is not included in the qos policies, vFC interfaces do not come up and increased drops occur.

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

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

Before You Begin

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **policy-map** [**type** {**network-qos** | **qos** | **queuing**}] *policy-name*	Creates a named object representing a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters. The three policy-map configuration modes are as follows: • network-qos—Network-wide (global) mode. CLI prompt: switch(config-pmap-nq)# • qos—Classification mode; this is the default mode. CLI prompt: switch(config-pmap-qos)# • queuing—Queuing mode. CLI prompt: switch(config-pmap-que)#
Step 3	switch(config)# **no policy-map** [**type** {**network-qos** | **qos** | **queuing**}] *policy-name*	(Optional) Deletes the specified policy map.
Step 4	switch(config-pmap)# **class** [**type** {**network-qos** | **qos** | **queuing**}] *class-name*	Associates a class map with the policy map, and enters configuration mode for the specified system class. The three class-map configuration modes are as follows: • network-qos—Network-wide (global) mode. CLI prompt: switch(config-pmap-c-nq)# • qos—Classification mode; this is the default mode. CLI prompt: switch(config-pmap-c-qos)# • queuing—Queuing mode. CLI prompt: switch(config-pmap-c-que)# **Note** The associated class map must be the same type as the policy-map type.
Step 5	switch(config-pmap)# **no class** [**type** {**network-qos** | **qos** | **queuing**}] *class-name*	(Optional) Deletes the class map association.

Configuring Type QoS Policies

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

Beginning with Cisco Release NX-OS Release 5.0(2)N1(1), for the Cisco Nexus 5548 switch, you can set a maximum of five qos groups for ingress traffic.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **policy-map type qos** *policy-name*
3. switch(config-pmap-qos)# **class type qos** *class-name*
4. switch(config-pmap-c-qos)# **set qos-group** *qos-group-value*
5. (Optional)   switch(config-pmap-c-qos)# **no set qos-group** *qos-group-value*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **policy-map type qos** *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	switch(config-pmap-qos)# **class type qos** *class-name*	Associates a class map with the policy map, and enters configuration mode for the specified system class. **Note** The associated class map must be the same type as the policy map type.
Step 4	switch(config-pmap-c-qos)# **set qos-group** *qos-group-value*	Configures one or more **qos-group** values to match on for classification of traffic into this class map. The list below identifies the ranges of the *qos-group-value*. There is no default value. For the Cisco Nexus 5020 switch and the Cisco Nexus 5010 switch, the range is from 2-5. For the Cisco Nexus 5548 switch, the range is from 1 to 5.
Step 5	switch(config-pmap-c-qos)# **no set qos-group** *qos-group-value*	(Optional) Removes the **qos-group** values from this class.

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

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

Configuring Type Network QoS Policies

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **policy-map type network-qos** *policy-name*
3. switch(config-pmap-nq)# **class type network-qos** *class-name*
4. switch(config-pmap-c-nq)# **mtu** *mtu-value*
5. (Optional) switch(config-pmap-c-nq)# **no mtu**
6. switch(config-pmap-c-nq)# **multicast-optimize**
7. (Optional) switch(config-pmap-c-nq)# **no multicast-optimize**
8. switch(config-pmap-c)# **pause no-drop** [**pfc-cos** *pfc-cos-value*]
9. (Optional) switch(config-pmap-c-nq)# **no pause no-drop**
10. switch(config-pmap-c-nq)# **queue-limit** *number-bytes* **bytes**
11. (Optional) switch(config-pmap-c-nq)# **no queue-limit** *number-bytes* **bytes**
12. switch(config-pmap-c-nq)# **set cos** *cos-value*
13. (Optional) switch(config-pmap-c-nq)# **no set cos** *cos-value*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **policy-map type network-qos** *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	switch(config-pmap-nq)# **class type network-qos** *class-name*	Associates a class map with the policy map, and enters configuration mode for the specified system class. **Note** The associated class map must be the same type as the policy map type.
Step 4	switch(config-pmap-c-nq)# **mtu** *mtu-value*	Specifies the MTU value in bytes. **Note** The *mtu-value* that you configure must be less than the value set by the **system jumbomtu** command.
Step 5	switch(config-pmap-c-nq)# **no mtu**	(Optional) Resets the MTU value in this class.
Step 6	switch(config-pmap-c-nq)# **multicast-optimize**	Enables multicast optimization. Multicast traffic in this class will be served by all available multicast queues. **Note** Only one class in a policy map can be configured for multicast optimization. **Note** For the Cisco Nexus 5548 switch, multicast optimization is enabled by default on class-default. You must remove it from class-default before enabling it on a user-defined class.

	Command or Action	**Purpose**
Step 7	switch(config-pmap-c-nq)# **no multicast-optimize**	(Optional) Disables multicast optimization.
Step 8	switch(config-pmap-c)# **pause no-drop** [**pfc-cos** *pfc-cos-value*]	Configures a no-drop class. If you do not specify this command, the default policy is drop. **Note** The operation for the drop policy is a simple tail drop, where arriving packets will be dropped if the queue increases to its allocated size. The *pfc-cos-value* range is from 0 to 7. This option is supported only for for a ACL-based system class (which filters traffic using criteria other than cos-based matches). **Caution** The list of CoS values can potentially include the CoS value that is used for FCoE traffic in class-fcoe. You must determine if this is desired behavior for your topology.
Step 9	switch(config-pmap-c-nq)# **no pause no-drop**	(Optional) Removes the no-drop option from this class.
Step 10	switch(config-pmap-c-nq)# **queue-limit** *number-bytes* **bytes**	Specifies the tail drop threshold on this interface. The threshold range is from 20480 to 204800 bytes. For the Cisco Nexus 5548 switch, the range is from 20480 to 426880 bytes. **Note** The queue limit can only be configured on drop classes. If you try to configure a queue limit on a no-drop class, or try to configure no-drop on a class where a queue limit is already defined, the CLI will return an error. Use the pause no-drop buffer size option to change the buffer size and pause threshold for the no-drop class.
Step 11	switch(config-pmap-c-nq)# **no queue-limit** *number-bytes* **bytes**	(Optional) Disables the queue limit specification in this class.
Step 12	switch(config-pmap-c-nq)# **set cos** *cos-value*	Specifies a 802.1Q CoS value which is used to mark packets on this interface. The value range is from 0 to 7.
Step 13	switch(config-pmap-c-nq)# **no set cos** *cos-value*	(Optional) Disables the marking operation in this class.

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

```
switch# configure terminal
switch(config)# policy-map type network-qos policy-que1
switch(config-pmap-nq)# class type network-qos class-que1
switch(config-pmap-c-nq)# mtu 5000
switch(config-pmap-c-nq)# pause no-drop pfc-cos 1-3,5
switch(config-pmap-c-nq)# set cos 4
```

Configuring Type Queuing Policies

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **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	switch(config-pmap-que)# **class type queuing** *class-name*	Associates a class map with the policy map, and enters configuration mode for the specified system class.
Step 4	switch(config-pmap-c-que)# **bandwidth percent** *percentage*	Specifies the guaranteed percentage of interface bandwidth allocated to this class. By default, no bandwidth is specified for a class. **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 5	switch(config-pmap-c-que)# **no bandwidth percent** *percentage*	(Optional) Removes the bandwidth specification from this class.
Step 6	switch(config-pmap-c-que)# **priority**	Specifies that traffic in this class is mapped to a strict priority queue. **Note** Only one class in each policy map can have strict priority set on it.
Step 7	switch(config-pmap-c-que)# **no priority**	(Optional) Removes the strict priority queuing from the traffic in this class.

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

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

Information About Marking

Marking is a method that you use to modify the QoS fields of the incoming and outgoing packets. The QoS fields that you can mark are CoS, IP precedence, and Differentiated Service Code Point (DSCP).

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

- DSCP (Supported only on the Cisco Nexus 5548 switch.)

- IP precedence

- CoS

Configuring DSCP Marking for the Cisco Nexus 5548 Switch

For the Cisco Nexus 5548 switch, 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 table below:

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

Table 4: Standard DSCP Values

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

Value	List of DSCP Values
af43	AF43 dscp (100110)—decimal value 38
cs1	CS1 (precedence 1) dscp (001000)—decimal value 8
cs2	CS2 (precedence 2) dscp (010000)—decimal value 16
cs3	CS3 (precedence 3) dscp (011000)—decimal value 24
cs4	CS4 (precedence 4) dscp (100000)—decimal value 32
cs5	CS5 (precedence 5) dscp (101000)—decimal value 40
cs6	CS6 (precedence 6) dscp (110000)—decimal value 48
cs7	CS7 (precedence 7) dscp (111000)—decimal value 56
default	Default dscp (000000)—decimal value 0
ef	EF dscp (101110)—decimal value 46

SUMMARY STEPS

1. **config t**
2. **policy-map type qos** *qos-policy-map-name*
3. **class** [**type qos**] {*class-map-name* | **class-default**}
4. **set dscp** *dscp-value*
5. **set qos-group** *y*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**config t**	Enters configuration mode.
Step 2	**policy-map type qos** *qos-policy-map-name*	Creates or accesses the policy map named policy-map-name, and then enters policy-map mode. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.

	Command or Action	Purpose
Step 3	**class** [**type qos**] {*class-map-name* | **class-default**}	Creates a reference to class-map-name, and enters policy-map class configuration mode. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**set dscp** *dscp-value*	Sets the DSCP value to dscp-value. See the Standards DSCP Values table.
Step 5	**set qos-group** *y*	Specifies the qos-group. The group value can be 1 - 5. **Note** Traffic in the class-default system class (qos-group 0), cannot be marked with DSCP.

This examples shows how to set the DSCP value to 10 and specify the qos-group to 2.

```
policy-map type qos test-bulkdata
   class type qos bulkdata
      set dscp 10
      set qos-group 2
```

Configuring IP Precedence Marking on the Cisco Nexus 5548 Switch

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. The following table shows the precedence values:

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

Table 5: Precedence Values

Value	List of Precedence Values
<0-7>	IP precedence value
critical	Critical precedence (5)
flash	Flash precedence (3)
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. **config t**
2. **policy-map** [**type qos**] *qos-policy-map-name*
3. **class** [**type qos**] {*class-map-name* | **class-default**}
4. **set precedence** *precedence-value*

DETAILED STEPS

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

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

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. **config t**
2. **policy-map** [**type network-qos**] *policy-map name*
3. **class** [**type network-qos**] {*class-map name* |**class-default**}
4. **set cos** *cos-value*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**config t**	Enters configuration mode.

	Command or Action	Purpose
Step 2	**policy-map** [**type network-qos**] *policy-map name*	Creates or accesses the policy map named policy-map-name, and then enters policy-map mode. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
Step 3	**class** [**type network-qos**] {*class-map name* |**class-default**}	Creates a reference to class-map-name, and enters policy-map class configuration mode. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 4	**set cos** *cos-value*	Sets the CoS value to cos-value. The value can range from 0 to 7. You can use this command only in egress policies.

Related Topics

QoS Configurations for Layer 3 Routing

Required CoS Marking Configuration in a Layer 3 Topology

Beginning in Cisco NX-OS Release 5.0(3)N1(1), in Layer 3 topologies you must configure each qos-group in the network-qos policy with a unique cos value.

SUMMARY STEPS

1. **show policy-map system**
2. **config t**
3. **policy-map** [**type network-qos**] *policy-map name*
4. **class** [**type network-qos**] {*class-map name* \|**class-default**}
5. **set cos** *cos-value*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show policy-map system**	Displays the already configured policy-maps and cos values.
		In Layer 3 topologies, each qos-group must have a unique cos value. Use the **show policy-map system** command to view cos values that have been used and that are unavailable for qos-groups.
Step 2	**config t**	Enters configuration mode.
Step 3	**policy-map** [**type network-qos**] *policy-map name*	Creates or accesses the policy map named policy-map-name, and then enters policy-map mode. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.

	Command or Action	**Purpose**
Step 4	**class** [**type network-qos**] {*class-map name* |**class-default**}	Creates a reference to class-map-name, and enters policy-map class configuration mode. Use the **class-default** keyword to select all traffic that is not currently matched by classes in the policy map.
Step 5	**set cos** *cos-value*	Sets the CoS value to cos-value. The value can range from 0 to 7. You can use this command only in egress policies.
		Note In Layer 3 topologies, each qos-group must have a unique cos configuration.

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

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

  policy-map type network-qos pn-01
    class type network-qos cn-01      match qos-group 1
      mtu 8500
      pause no-drop
      set cos 2
    class type network-qos cn-02      match qos-group 2
      set cos 4
      mtu 9216
    class type network-qos cn-03      match qos-group 3
      mtu 8000
      set cos 6
    class type network-qos cn-04      match qos-group 4
      mtu 8750
      set cos 7
    class type network-qos cn-ip-multicast      match qos-group 5
      set cos 5
      mtu 7500
    class type network-qos class-default      match qos-group 0
      mtu 1500
      multicast-optimize
      set cos 1
...
switch# configure terminal switch(config)# policy-map type network-qos pn-01
switch(config-pmap-nq)# class type network-qos cn-05
switch(config-pmap-c-nq)# set cos 3
```

Configuring Layer 3 Multicast Queuing

Beginning with NX-OS Release 5.0(3)N1(1), on Cisco Nexus 5000 Platform switches, you can map CoS values to an egress queue. You can have a maximum of 4 multicast queues for Layer 3 multicast traffic.

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **wrr-queue cos-map** *queue-id cos1 ...cos8*

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **wrr-queue cos-map** *queue-id cos1 ...cos8*	Maps the assigned CoS values to an egress queue.
		The egress queue range is from 1 to 4, where 4 can be configured as the expedite queue.
		You can enter up to eight CoS values. Separate each value with a space. The range is from 0 to 7.
		The defaults are as follows:
		• Receive queue 0 and transmit queue 0: CoS 0 and 1.
		• Receive queue 1 and transmit queue 1: CoS 2 and 3.
		• Receive queue 2 and transmit queue 2: CoS 4 and 5.
		• Receive queue 3 and transmit queue 3: CoS 6 and 7.

This example shows how to configure a Layer 3 interface:

```
switch# configure terminal
switch(config)# wrr-queue cos-map 1 0 1
switch(config)#
```

Configuring a Service Policy for a Layer 3 Interface

Beginning with NX-OS Release 5.0(3)N1(1), on Cisco Nexus 5000 Platform switches, you can configure a service policy to a Layer 3 interface.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface ethernet** *slot*/*port*
3. switch(config-if)# **no switchport**
4. switch(config-if)# **service-policy** [**type** {**qos** | **queuing**} [**input** | **output**]*policy-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **interface ethernet** *slot*/*port*	Enters configuration mode for the specified interface.
Step 3	switch(config-if)# **no switchport**	Selects the Layer 3 interface.

	Command or Action	Purpose
Step 4	switch(config-if)# **service-policy** [**type** {**qos** | **queuing**} [**input** | **output**]*policy-name*	Specifies the policy map to use as the service policy for the Layer 3 interface. There are two policy-map configuration modes: • qos—Classification mode; this is the default mode. • queuing—Queuing mode. **Note** The **input** keyword specifies that this policy map should be applied to traffic received on an interface. The **output** keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply **input** to a qos policy; you can apply both **input** and **output** to a queuing policy.

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

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

Changing the Bandwidth Allocated to Unicast and Multicast Traffic

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

,

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface ethernet** *slot*/*port*
3. switch(config-if)# **wrr unicast-bandwidth** percentage-value

DETAILED STEPS

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

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

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

Attaching the System Service Policy

You can use the **service-policy** command to associate the system class policy map as the service policy for the system.

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **system qos**	Enters system class configuration mode.
Step 3	switch(config-sys-qos)# **service-policy type** {**network-qos**	**qos**	**queuing**} [**input**	**output**] *policy-name*	Specifies the policy map to use as the service policy for the system. There are three policy-map configuration modes: • network-qos—Network-wide (system qos) mode. • qos—Classification mode (system qos input or interface input only). • queuing—Queuing mode (input and output at system qos and interface). **Note** There is no default policy-map configuration mode; you must specify the **type**. The **input** keyword specifies that this policy map should be applied to traffic received on an interface. The **output** keyword specifies that this policy-map should be applied to traffic transmitted from an interface. You can only apply **input** to a qos policy; you can apply both **input** and **output** to a queuing policy.
Step 4	switch(config-sys-qos)# **service-policy type** {**network-qos**	**qos**	**queuing**} [**input**	**output**] *fcoe default policy-name*	(Optional) Specifies the default FCoE policy map to use as the service policy for the system. There are four pre-defined policy-maps for FCoE: • service-policy type qos input fcoe-default-in-policy • service-policy type queuing input fcoe-default-in-policy • service-policy type queuing output fcoe-default-out-policy • service-policy type network-qos fcoe-default-nq-policy **Note** Befoe enabling FCoE on the Cisco Nexus 5548 switch, you must attach the pre-defined FCoE policy maps to the type qos, type network-qos, and type queuing policy maps.

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

```
switch(config)# class-map type network-qos ethCoS4
switch(config-cmap-nq)# match qos-group
switch(config-cmap-nq)# exit
switch(config)# policy-map type network-qos ethNoDrop
switch(config-pmap-nq)# class type network-qos ethCoS4
switch(config-pmap-c-nq)# pause no-drop
switch(config-pmap-c-nq)# exit
switch(config-pmap-nq)# exit
switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos ethNoDrop
```

Restoring the Default System Service Policies

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **system qos**	Enters system class configuration mode.
Step 3	switch(config-sys-qos)# **no service-policy type qos input** *policy-map name*	Resets the classification mode policy map. This policy-map configuration is for system qos input or interface input only:
Step 4	switch(config-sys-qos)# **no service-policy type network-qos** *policy-map name*	Resets the network-wide policy map.
Step 5	switch(config-sys-qos)# **no service-policy type queuing output** *policy-map name*	Resets the output queuing mode policy map.
Step 6	switch(config-sys-qos)# **no service-policy type queuing input** *policy-map name*	Resets the input queuing mode policy map.

This example shows how to reset the system qos configuration:

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

```
switch(config-sys-qos)# no service-policy type queuing output my-out-policy
switch(config-sys-qos)# no service-policy type queuing input my-in-policy
```
This example shows the default service policies:

```
switch# show policy-map


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

  policy-map type qos default-in-policy
    class type qos class-fcoe
      set qos-group 1
    class type qos class-default
      set qos-group 0

  Type queuing policy-maps
  ========================
  policy-map type queuing default-in-policy
   class type queuing class-fcoe
      bandwidth percent 50
    class type queuing class-default
      bandwidth percent 50
 policy-map type queuing default-out-policy
    class type queuing class-fcoe
      bandwidth percent 50
    class type queuing class-default
      bandwidth percent 50

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

  policy-map type network-qos default-nq-policy
    class type network-qos class-fcoe
      pause no-drop
      mtu 2240
    class type network-qos class-default
      mtu 1538
```

Enabling the Jumbo MTU

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

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

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

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

Verifying the Jumbo MTU

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

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

```
switch# show interface ethernet 1/2
Ethernet1/2 is up
...
    Rx
    1547805598 Input Packets 1547805596 Unicast Packets 0 Multicast Packets
    0 Broadcast Packets 1301767362 Jumbo Packets 33690 Storm Suppression Packets
    7181776513802 Bytes
    Tx
    1186564478 Output Packets 7060 Multicast Packets
    0 Broadcast Packets 997813205 Jumbo Packets
    4813632103603 Bytes
...
```

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

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

Configuring QoS on Interfaces

Configuring Untagged CoS

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

Beginning in Cisco NX-OS Release 5.0(3)N1(1), you can configure flow control on a Layer 2 or Layer 3 interface. Use the **no switchport** command to configure a Layer 3 interface

Beginning with Cisco NX-OS Release 5.0(2)N1(1), on the Cisco Nexus 5548 switch, you can configure a type qos policy map and untagged CoS on the same interface.

Note   On the Cisco Nexus 5010 switch and the Cisco Nexus 5020 switch, untagged CoS and type QoS input policies are mutually exclusive on an Ethernet or EtherChannel interface. If a type qos policy is configured at the interface, untagged frames received over that interface will not match any **match cos 0** commands in the policy.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** {**ethernet** [*chassis/*]*slot/port* | **port-channel** *channel-number*}
3. (Optional)   switch(config-if)# **no switchport**
4. switch(config-if)# **untagged cos** *cos-value*

DETAILED STEPS

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

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

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# untagged cos 4
```
This example shows how to set the CoS value to 3 for untagged frames received on a Layer 3 interface:

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

Configuring Interface Service Policy

An input qos policy is a service policy applied to incoming traffic on an Ethernet interface for classification. For type queuing, the output policy is applied to all outgoing traffic that matches the specified class. When you configure an input queuing policy on an interface or EtherChannel, the switch sends the configuration data to the adapter using the DCBX protocol.

> **Note** Type qos policies can be activated only on Cisco Nexus 5000 Series interfaces and Cisco Nexus 2000 Series Fabric Extender interfaces. Type qos policies on Fabric Extender fabric interfaces or Fabric Extender fabric EtherChannel interfaces are ineffective, though the Cisco NX-OS CLI does not reject the configuration.
>
> We recommend that you do not configure type qos policy-maps on Fabric Extender fabric interfaces or Fabric Extender fabric EtherChannel interfaces to avoid wasting hardware resources.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** {**ethernet** [*chassis/*]*slot/port* | **port-channel** *channel-number*}
3. switch(config-if)# **service-policy** [**type** {**qos** | **queuing**}] [**input** | **output**]*policy-name*
4. switch(config-if)# **service-policy input** *policy-name*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **interface** {**ethernet** [*chassis/*]*slot/port*	**port-channel** *channel-number*}	Enters configuration mode for the specified interface. **Note** The service policy on a port channel applies to all member interfaces.
Step 3	switch(config-if)# **service-policy** [**type** {**qos**	**queuing**}] [**input**	**output**]*policy-name*	Specifies the policy map to use as the service policy for the system. There are two policy-map configuration modes: • qos—Classification mode; this is the default mode. • queuing—Queuing mode. **Note** The **input** keyword specifies that this policy map should be applied to traffic received on an interface. The **output** keyword specifies that this policy map should be applied to traffic transmitted from an interface. You can only apply **input** to a qos policy; you can only apply both **input** and **output** to a queuing policy.
Step 4	switch(config-if)# **service-policy input** *policy-name*	Applies the policy map to the interface. **Note** There is a restriction that system type qos policy cannot be the same as any the type qos policy applied to an interface or EtherChannel.

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

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

Disabling Slow Port Pruning on Multicast Traffic on the Cisco Nexus 5548 Switch

You can disable slow port pruning on multicast packets.

An interface on the Cisco Nexus 5548 switch can become congested when it receives excessive multicast traffic or when the mixed unicast and multicast traffic rate exceeds the port bandwidth. When multiple interfaces receive the same multicast flow and one or more ports experience congestion, the slow port prunning feature allows the switch to drop only the multicast packets for the congested port. This feature is turned on by default. To turn the slow port pruning feature off, enter the **hardware multicast disable-slow-port-pruning** command.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **hardware multicast disable-slow-port-pruning**
3. switch(config)# **no hardware multicast disable-slow-port-pruning**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **hardware multicast disable-slow-port-pruning**	Disables slow port pruning on multicast packets. The default is enabled.
Step 3	switch(config)# **no hardware multicast disable-slow-port-pruning**	Enables the slow port pruning feature.

This example shows how to disable slow port pruning on the Cisco Nexus 5548 switch:
```
switch(config)# hardware multicast disable-slow-port-pruning
switch(config)#
```

Verifying QoS Configuration

To verify QoS configuration information, perform one of these tasks:

Command	Purpose
switch# **show class-map**	Displays the class maps defined on the switch.
switch# **show policy-map** [*name*]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
switch# **show policy-map interface** [*interface number*]	Displays the policy map settings for an interface or all interfaces.

Command	Purpose
switch# **show policy-map system**	Displays the policy map settings attached to the system qos.
switch# **show policy-map type** {**network-qos** | **qos** | **queuing**} [*name*]	Displays the policy map settings for a specific policy type. Optionally, you can display the named policy only.
switch# **show queuing interface** [*interface slot/port*]	Displays the queue configuration and statistics.
switch# **show interface flowcontrol** [**module** *number*]	Displays the detailed listing of the flow control settings on all interfaces.
switch# **show interface** [*interface slot/port*] **priority-flow-control** [**module** *number*]	Displays the priority flowcontrol details for a specified interface.
switch# **show interface untagged-cos** [**module** *number*]	Displays the untagged CoS values for all interfaces.
switch# **show wrr-queue cos-map** [*var*]	Displays the mapped CoS values to egress queues.
switch# **running-config ipqos**	Displays information about the running configuration for QoS.
switch# **startup-config ipqos**	Displays information about the startup configuration for QoS.

Examples

This example shows how to display all type queuing class maps on a switch that runs Cisco NX-OS Release 5.0(3)N1(1):

```
switch# show class-map type queuing
Type queuing class-maps
=======================
class-map type queuing class-fcoe
match qos-group 1
class-map type queuing class-default
match qos-group 0
class-map type queuing class-all-flood
match qos-group 2
class-map type queuing class-ip-multicast
match qos-group 2
switch#
```

This example shows how to display the flow control settings on Layer 3 interfaces where Ethernet 1/5 is configured as a routed interface:

```
switch# show interface flowcontrol
Port    Send    FlowControl Receive FlowControl RxPause TxPause
        admin    oper       admin    oper
--------------------------------------------------------------------------
Eth1/1  off      off        off      off            0       0
Eth1/2  off      off        off      off            0       0
Eth1/3  off      off        off      off            0       0
Eth1/4  off      off        off      off            0       0
Eth1/5  off      off        on       on             0       0
Eth1/6  off      off        off      off            0       0
Eth1/7  off      off        off      off            0       0
Eth1/8  off      off        off      off            0       0
```

```
Eth1/9   off      off      off      off         0        0
Eth1/10  off      off      off      off         0        0
Eth1/11  off      off      off      off         0        0
Eth1/12  off      off      off      off         0        0
Eth1/13  off      off      off      off         0        0
Eth1/14  off      off      off      off         0        0
<--Output truncated-->
switch#
--------------------------------------------------------------------------------
```

This example shows how to display the priority flow control information for a specified Layer 3 interface:

```
switch# show interface ethernet 1/5 priority-flow-control
============================================================
Port            Mode Oper(VL bmap) RxPPP TxPPP
============================================================
Ethernet1/5    On   On   (0) 0 0
switch#
```

This example shows how to display the untagged CoS values for all nterfaces (Layer 2, Layer 3) on a switch running Cisco NX-OS Release 5.0(3)N1(1):

```
switch# show interface untagged-cos
S3(config-if)# show int untagged-cos
================================
Interface Untagged-CoS
================================
port-channel100
port-channel127
port-channel128
Ethernet1/1
Ethernet1/2
Ethernet1/3
Ethernet1/4
Ethernet1/5  3
Ethernet1/6
Ethernet1/7
Ethernet1/8
Ethernet1/9
Ethernet1/10
Ethernet1/11
Ethernet1/12
:
<--snip-->
Ethernet3/31
Ethernet3/32
Ethernet100/1/1
Ethernet100/1/2
<--Output truncated-->
switch#
```

This example shows how to display QoS information on a switch that runs Cisco NX-OS Release 5.0(3)N1(1):

```
switch# show running-config ipqos
!Command: show running-config ipqos
!Time: Sun Apr 20 07:22:36 2008

version 5.0(3)N1(1)
class-map type qos class-fcoe
class-map type qos match-all c1
  match cos 3-5
class-map type queuing class-fcoe
  match qos-group 1
class-map type queuing class-all-flood
  match qos-group 2
class-map type queuing class-ip-multicast
  match qos-group 2
policy-map type qos p1
  class c1
    set qos-group 3
class-map type network-qos n1
 match qos-group 1
class-map type network-qos class-fcoe
  match qos-group 1
class-map type network-qos class-all-flood
```

```
   match qos-group 2
class-map type network-qos class-ip-multicast
   match qos-group 2
policy-map type network-qos n2
   class type network-qos n1
   class type network-qos class-default
    multicast-optimize

interface Ethernet1/5
   priority-flow-control mode on
   untagged cos 3
switch#

The above display shows the QoS information for Layer 3 interfaces (Ethernet 1/5 configured
 as a Layer
3 interface).
```

This example shows how to display the CoS values that are mapped to the egress queue:

```
switch# show wrr-queue cos-map
MCAST Queue ID     Cos Map
0                    0 1
1                    2
2                    3 4 5
3                    6 7
switch#
```

This example shows how to display the class maps defined on the switch:

```
switch# show class-map


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

    class-map type qos c1
      match cos 0,7

    class-map type qos c2
      match protocol ldp
      match ip rtp 2000-65535
      match dscp 10,12
      match precedence 6-7
      match protocol dhcp
      match protocol arp

    class-map type qos c3
      match cos 2,4-6

    class-map type qos c4
      match access-group name ipv4

    class-map type qos class-fcoe
      match cos 3

    class-map type qos class-default
      match any

    class-map type qos class-ip-multicast
      match ip multicast


  Type queuing class-maps
  =======================

    class-map type queuing c1
      match qos-group 2

    class-map type queuing c2
      match qos-group 3

    class-map type queuing c3
      match qos-group 4
```

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

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


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

  class-map type network-qos c1
    match qos-group 2

  class-map type network-qos c2
    match qos-group 3

  class-map type network-qos c3
    match qos-group 4

  class-map type network-qos c4
    match qos-group 5

  class-map type network-qos class-fcoe
    match qos-group 1

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

This example shows how to display the policy maps defined on the switch:

```
switch# show policy-map


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

  policy-map type qos p1
    class type qos c1
      set qos-group 2
    class type qos c3
      set qos-group 4
    class type qos c4
      set qos-group 5
    class type qos c2
      set qos-group 3
    class type qos c22
      set qos-group 3
    class type qos class-fcoe
      set qos-group 1
    class type qos class-default
      set qos-group 0


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

  policy-map type queuing p1
    class type queuing c2
      bandwidth percent 10
    class type queuing c4
      bandwidth percent 25
    class type queuing c1
      bandwidth percent 20
    class type queuing c3
      bandwidth percent 5
    class type queuing class-fcoe
      bandwidth percent 30
    class type queuing class-default
      bandwidth percent 10
```

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

policy-map type network-qos p1
  class type network-qos c1
    mtu 5000
  class type network-qos c2
    mtu 9216
    queue-limit 30000 bytes
  class type network-qos c3
    mtu 8000
  class type network-qos c4
    pause no-drop
  class type network-qos class-fcoe
    pause no-drop
    mtu 2240
  class type network-qos class-default
    mtu 1538
```

This example shows how to display the policy maps attached on the system qos:

```
switch# show policy-map system


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

  policy-map type network-qos p1
    class type network-qos c1       match qos-group 2

      mtu 5000
    class type network-qos c2       match qos-group 3

      mtu 9216
      queue-limit 30000 bytes
    class type network-qos c3       match qos-group 4

      mtu 8000
    class type network-qos c4       match qos-group 5

      pause no-drop
    class type network-qos class-fcoe       match qos-group 1

      pause no-drop
      mtu 2240
    class type network-qos class-default       match qos-group 0

      mtu 1538


  Service-policy (queuing) input:   p1
    policy statistics status:   disabled

    Class-map (queuing):   c2 (match-any)
      Match: qos-group 3
      bandwidth percent 10

    Class-map (queuing):   c4 (match-any)
      Match: qos-group 5
      bandwidth percent 25

    Class-map (queuing):   c1 (match-any)
      Match: qos-group 2
      bandwidth percent 20

    Class-map (queuing):   c3 (match-any)
      Match: qos-group 4
      bandwidth percent 5

    Class-map (queuing):   class-fcoe (match-any)
      Match: qos-group 1
      bandwidth percent 30
```

```
      Class-map (queuing):   class-default (match-any)
        Match: qos-group 0
        bandwidth percent 10

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

    Class-map (queuing):   class-fcoe (match-any)
      Match: qos-group 1
      bandwidth percent 50

    Class-map (queuing):   class-default (match-any)
      Match: qos-group 0
      bandwidth percent 50
```

This example shows how to display the policy maps attached to an interface:

```
switch# show policy-map interface ethernet 1/1

Global statistics status :   disabled

Ethernet1/1

  Service-policy (qos) input:   p2
    policy statistics status:   disabled

    Class-map (qos):   c1 (match-any)
      Match: cos 0,7
      set qos-group 2

    Class-map (qos):   c2 (match-any)
      Match: protocol ldp
      Match: ip rtp 2000-65535
      Match: dscp 10,12
      Match: precedence 6-7
      Match: protocol dhcp
      Match: protocol arp
      set qos-group 3

    Class-map (qos):   c3 (match-any)
      Match: cos 2,4-6
      set qos-group 4

    Class-map (qos):   class-ip-multicast (match-any)
      Match: ip multicast
      set qos-group 5
    Class-map (qos):   class-fcoe (match-any)
      Match: cos 3
      set qos-group 1

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

  Service-policy (queuing) input:   p1
    policy statistics status:   disabled

    Class-map (queuing):   c2 (match-any)
      Match: qos-group 3
      bandwidth percent 10

    Class-map (queuing):   c4 (match-any)
      Match: qos-group 5
      bandwidth percent 25

    Class-map (queuing):   c1 (match-any)
      Match: qos-group 2
      bandwidth percent 20
```

```
      Class-map (queuing):   c3 (match-any)
        Match: qos-group 4
        bandwidth percent 5

      Class-map (queuing):   class-fcoe (match-any)
        Match: qos-group 1
        bandwidth percent 30

      Class-map (queuing):   class-default (match-any)
        Match: qos-group 0
        bandwidth percent 10

   Service-policy (queuing) output:   p2
     policy statistics status:   disabled

      Class-map (queuing):   c1 (match-any)
        Match: qos-group 2
        bandwidth percent 5
        priority

      Class-map (queuing):   c2 (match-any)
        Match: qos-group 3
        bandwidth percent 20

      Class-map (queuing):   c3 (match-any)
        Match: qos-group 4
        bandwidth percent 20

      Class-map (queuing):   c4 (match-any)
        Match: qos-group 5
        bandwidth percent 40

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

      Class-map (queuing):   class-default (match-any)
        Match: qos-group 0
        bandwidth percent 5
```

This example shows how to display the queue configuration and statistics:

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

Interface Ethernet1/1 TX Queuing
qos-group   sched-type   oper-bandwidth
    0       WRR                 5
    1       WRR                10
    2       priority            5
    3       WRR                20
    4       WRR                20
    5       WRR                40

Interface Ethernet1/1 RX Queuing
qos-group  0:
    q-size: 21120, MTU: 1538
    drop-type: drop, xon: 0, xoff: 132
    Statistics:
        Pkts received over the port            : 1265258330
        Ucast pkts sent to the cross-bar       : 0
        Mcast pkts sent to the cross-bar       : 2883444539
        Ucast pkts received from the cross-bar : 0
        Pkts sent to the port                  : 367529517
        Pkts discarded on ingress              : 781087 (36419)
        Per-priority-pause status              : Rx (Inactive), Tx (Inactive)
```

```
        qos-group  1:
            q-size: 76800, MTU: 2240
            drop-type: no-drop, xon: 128, xoff: 240
            Statistics:
                Pkts received over the port           : 0
                Ucast pkts sent to the cross-bar      : 0
                Mcast pkts sent to the cross-bar      : 0
                Ucast pkts received from the cross-bar  : 0
                Pkts sent to the port                 : 0
                Pkts discarded on ingress             : 0 (0)
                Per-priority-pause status             : Rx (Inactive), Tx (Inactive)

        qos-group  2:
            q-size: 20480, MTU: 5000
            drop-type: drop, xon: 0, xoff: 128
            Statistics:
                Pkts received over the port           : 0
                Ucast pkts sent to the cross-bar      : 0
                Mcast pkts sent to the cross-bar      : 0
                Ucast pkts received from the cross-bar  : 0
                Pkts sent to the port                 : 0
                Pkts discarded on ingress             : 0 (0)
                Per-priority-pause status             : Rx (Inactive), Tx (Inactive)

        qos-group  3:
            q-size: 30080, MTU: 9216
            drop-type: drop, xon: 0, xoff: 188
            Statistics:
                Pkts received over the port           : 0
                Ucast pkts sent to the cross-bar      : 0
                Mcast pkts sent to the cross-bar      : 0
                Ucast pkts received from the cross-bar  : 0
                Pkts sent to the port                 : 0
                Pkts discarded on ingress             : 0 (0)
                Per-priority-pause status             : Rx (Inactive), Tx (Inactive)

        qos-group  4:
            q-size: 20480, MTU: 8000
            drop-type: drop, xon: 0, xoff: 128
            Statistics:
                Pkts received over the port           : 0
                Ucast pkts sent to the cross-bar      : 0
                Mcast pkts sent to the cross-bar      : 0
                Ucast pkts received from the cross-bar  : 0
                Pkts sent to the port                 : 0
                Pkts discarded on ingress             : 0(0)
                Per-priority-pause status             : Rx (Inactive), Tx (Inactive)

        qos-group  5:
            q-size: 76800, MTU: 1538
            drop-type: no-drop, xon: 128, xoff: 242
            Statistics:
                Pkts received over the port           : 0
                Ucast pkts sent to the cross-bar      : 0
                Mcast pkts sent to the cross-bar      : 0
                Ucast pkts received from the cross-bar  : 0
                Pkts sent to the port                 : 0
                Pkts discarded on ingress             : 0(0)
                Per-priority-pause status             : Rx (Inactive), Tx (Inactive)

    Total Multicast crossbar statistics:
        Mcast pkts received from the cross-bar       : 557155515
```

Example QoS Configurations

QoS Example 1

This example shows how to configure traffic in the entire system matching an access control list to have the frame CoS fields rewritten to the value 5.

SUMMARY STEPS

1. Set up the ingress classification policy (the access control list was defined previously).
2. Attach the classification policy to the system.
3. Set up the system class allocation and rewrite policy. Allocate the system class for qos-group 4 and define the rewrite action.
4. Attach the allocation and rewrite policy to the system.

DETAILED STEPS

	Command or Action	Purpose

Step 1	Set up the ingress classification policy (the access control list was defined previously).	`(config)# class-map type qos cmap-qos-acl`
		`(config-cmap-qos)# match access-group ACL-CoS`
		`(config-cmap-qos)# exit`
		`(config)# policy-map type qos pmap-qos-acl`
		`(config-pmap-qos)# class cmap-qos-acl`
		`(config-pmap-c-qos)# set qos-group 4`
		`(config-pmap-c-qos)# exit`
		`(config-pmap-qos)# exit`

Step 2	Attach the classification policy to the system.	`(config)# system qos`
		`(config-sys-qos)# service-policy type qos input pmap-qos-acl`
		`(config-sys-qos)# exit`

Step 3	Set up the system class allocation and rewrite policy. Allocate the system class for qos-group 4 and define the rewrite action.	`(config)# class-map type network-qos cmap-nq-acl`
		`(config-cmap-nq)# match qos-group 4`
		`(config-cmap-nq)# exit`
		`(config)# policy-map type network-qos pmap-nq-acl`
		`(config-pmap-nq)# class type network-qos cmap-nq-acl`
		`(config-pmap-c-nq)# set cos 5`
		`(config-pmap-c-nq)# exit`
		`(config-pmap-nq)# exit`

Step 4	Attach the allocation and rewrite policy to the system.	`(config)# system qos`
		`(config-sys-qos)# service-policy type network-qos pmap-nq-acl`
		`(config-sys-qos)# exit`

QoS Example 2

This example shows how to use an access control list to apply 50% bandwidth to traffic on Ethernet interface 1/3 that matches traffic on Ethernet interface 1/1.

SUMMARY STEPS

1. Set up the ingress classification policy.
2. Attach the classification policy to the interface Ethernet 1/1.
3. Set up the system-wide definition of the qos-group first.
4. Set up the egress bandwidth policy.
5. Attach the bandwidth policy to the egress interface.
6. Allocate the system class for qos-group 2.
7. Set up the network-qos policy.
8. Attach the network-qos policy to the system.

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	Set up the ingress classification policy.	`(config)# ` **`class-map type qos cmap-qos-bandwidth`** `(config-cmap-qos)# ` **`match access-group ACL-bandwidth`** `(config-cmap-qos)# ` **`exit`** `(config)# ` **`policy-map type qos pmap-qos-eth1-1`** `(config-pmap-qos)# ` **`class cmap-qos-bandwidth`** `(config-pmap-c-qos)# ` **`set qos-group 2`** `(config-pmap-c-qos)# ` **`exit`** `(config-pmap-qos)# ` **`exit`**
Step 2	Attach the classification policy to the interface Ethernet 1/1.	`(config)# ` **`interface ethernet 1/1`** `(config-if)# ` **`service-policy type qos input pmap-qos-eth1-1`** `(config-if)# ` **`exit`**
Step 3	Set up the system-wide definition of the qos-group first.	`(config)# ` **`class-map type queuing cmap-que-bandwidth`** `(config-cmap-que)# ` **`match qos-group 2`** `(config-cmap-que)# ` **`exit`**
Step 4	Set up the egress bandwidth policy.	**Note** Before you can successfully allocate bandwidth to the user-defined class cmap-que-bandwidth, you must first reduce the default bandwidth configuration on class-default and class-fcoe. `(config)# ` **`policy-map type queuing pmap-que-eth1-2`** `(config-pmap-que)# ` **`class type queuing class-default`** `(config-pmap-c-que)# ` **`bandwidth percent 10`** `(config-pmap-c-que)# ` **`exit`** `(config-pmap-que)# ` **`class type queuing class-fcoe`** `(config-pmap-c-que)# ` **`bandwidth percent 40`** `(config-pmap-c-que)# ` **`exit`** `(config-pmap-que)# ` **`class type queuing cmap-que-bandwidth`** `(config-pmap-c-que)# ` **`bandwidth percent 50`** `(config-pmap-c-que)# ` **`exit`** `(config-pmap-que)# ` **`exit`**

	Command or Action	**Purpose**

Step 5	Attach the bandwidth policy to the egress interface.	`(config)# interface ethernet 1/3`
		`(config-if)# service-policy type queuing output pmap-que-eth1-2`
		`(config-if)# exit`

Step 6	Allocate the system class for qos-group 2.	`(config)# class-map type network-qos cmap-nq-bandwidth`
		`(config-cmap-nq)# match qos-group 2`
		`(config-cmap-nq)# exit`

Step 7	Set up the network-qos policy.	`(config)# policy-map type network-qos pmap-nq-bandwidth`
		`(config-pmap-nq)# class type network-qos cmap-nq-bandwidth`
		`(config-pmap-c-nq)# exit`
		`(config-pmap-nq)# exit`

Step 8	Attach the network-qos policy to the system.	`(config)# system qos`
		`(config-sys-qos)# service-policy type network-qos pmap-nq-bandwidth`
		`(config-sys-qos)# exit`

QoS Example 3

This example shows how to attach a 802.1p tag with a CoS value of 3 to incoming untagged packets, and force priority-flow-control negotiation on Ethernet interface 1/15.

SUMMARY STEPS

1. Set up the ingress classification policy (the access control list was defined previously).

DETAILED STEPS

	Command or Action	**Purpose**

Step 1	Set up the ingress classification policy (the access control list was defined previously).	`(config)# interface Ethernet 1/15`
		`(config-if)# untagged cos 3`
		`(config-if)# priority-flow-control mode on`
		`(config-if)# exit`