Cisco Nexus 5600 Series NX-OS Quality of Service Configuration Guide, Release 7.x

First Published: 2014-03-17

Last Modified: 2019-07-16

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

Text Part Number: OL-31638-03

CONTENTS

Preface

The preface contains the following sections:

Audience

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

Document Conventions

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

Command descriptions use the following conventions:

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

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

Examples use the following conventions:

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

This document uses the following conventions:

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

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

Documentation Feedback

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

We appreciate your feedback.

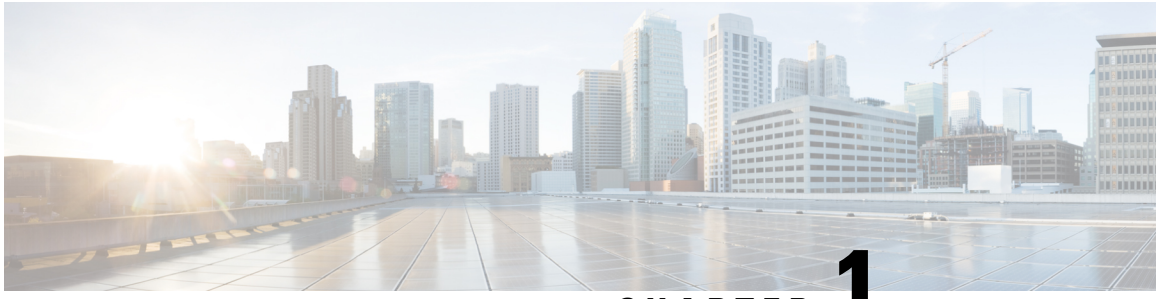
Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.

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

Cisco Bug Search Tool

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



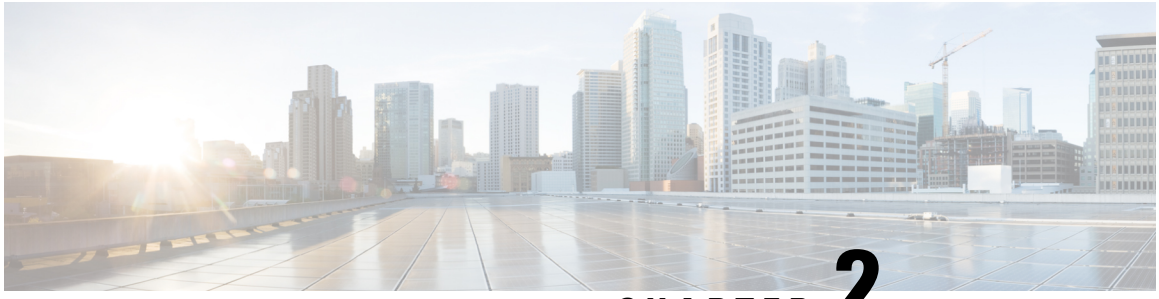
CHAPTER 1

New and Changed Information

New and Changed Information

This chapter provides release-specific information for each new and changed feature in the *Cisco Nexus 5600 Series NX-OS QoS Configuration Guide , Release 7.x.*

Feature	Description	Changed in Release	Where Documented
FEX-Based ACL Classification	The FEX-Based ACL Classification feature uses TCAM resources on a FEX to perform ACL-based packet classification of incoming packets on the switch.	7.0(3)N1(1)	Configuring FEX-Based ACL Classification, on page 105
Buffer Utilization Histogram	The Buffer Utilization Histogram feature allows you to analyze the maximum queue depths and buffer utilization in the system in real time. Instantaneous or real time buffer utilization information is supported.	7.0(2)N1(1)	Configuring Buffer Utilization Histogram, on page 97



CHAPTER 2

Overview

This chapter contains the following sections:

Information About Quality of Service

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

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

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

Modular QoS CLI

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

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

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

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

class-map

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

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

policy-map

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

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

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

network-qos

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

qos

Defines MQC objects that you can use for classification.

queuing

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

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

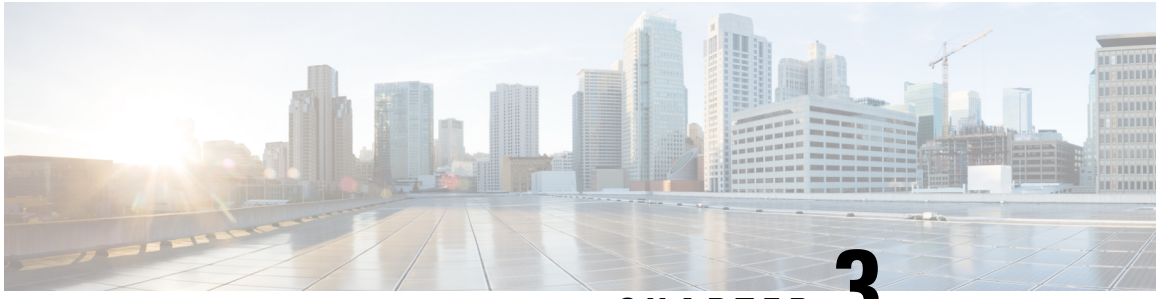
You can attach policies to interfaces or EtherChannels as well as at the global system level by using the **service-policy** command.

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

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

QoS for Traffic Directed to the CPU

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



CHAPTER 3

Configuring Classification

This chapter contains the following sections:

Information About Classification

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

You can create class maps to represent each traffic class by matching packet characteristics with classification criteria.

Table 1: Classification Criteria

Classification Criteria	Description
Class map	Criteria specified in a named class-map object.
Precedence	Precedence value within the Type of Service (ToS) byte of the IP Header.
Differentiated Services Code Point (DSCP)	DSCP value within the DIffServ field of the IP Header.
Protocol	Selected set of protocols, including Address Resolution Protocol (ARP) and Connectionless Network Service (CLNS).
IP RTP	Identify applications using Real-time Transport Protocol (RTP) by UDP port number range.
ACL	Traffic is classified by the criteria defined in the access control list (ACL).

Table 2: Supported RFCs

RFC	Title
RFC 2474	Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers

Ingress Classification Policies

You use classification to partition traffic into classes. You classify the traffic based on the packet property (CoS field) or the packet header fields that include IP precedence, Differentiated Services Code Point (DSCP), and Layer 2 to Layer 4 parameters. The values used to classify traffic are called match criteria.

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

Licensing Requirements for Classification

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Configuring Classification

Configuring Class Maps

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

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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:

	Command or Action	Purpose
		• **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	(Optional) switch(config)# **class-map** [**type qos**] [**match-all** | **match-any**] *class-map name*	Specifies that packets must match any or all criteria that is defined for a class map. • **match-all**—Classifies traffic if packets match all criteria that is defined for a specified class map (for example, if both the defined CoS and the ACL criteria match). • **match-any**—Classifies traffic if packets match any criteria that is defined for a specified class map (for example, if either the CoS or the ACL criteria matches). Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 4	(Optional) switch(config)# **no class-map** [**type** {**network-qos** | **qos** | **queuing**}] *class-name*	Deletes the specified class map. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.

Configuring CoS Classification

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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,

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

Example

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

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

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

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

Configuring Precedence Classification

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

Table 3: Precedence Values

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **class-map type qos match-any** *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. For a list of precedence values, see the Precedence Values table.
Step 4	(Optional) switch((config-cmap-qos)# **no match precedence** *precedence-values*	Removes the match from the traffic class. For a list of precedence values, see the Precedence Values table.

Example

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

```
switch# configure terminal
switch(config)# class-map type qos match-any class_precedence
switch(config-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 DSCP Classification

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

Table 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

Value	List of DSCP Values
af31	AF31 dscp (011010)—decimal value 26
af32	AF32 dscp (011100)—decimal value 28
af33	AF33 dscp (011110)—decimal value 30
af41	AF41 dscp (100010)—decimal value 34
af42	AF42 dscp (100100)—decimal value 36
af43	AF43 dscp (100110)—decimal value 38
cs1	CS1 (precedence 1) dscp (001000)—decimal value 8
cs2	CS2 (precedence 2) dscp (010000)—decimal value 16
cs3	CS3 (precedence 3) dscp (011000)—decimal value 24
cs4	CS4 (precedence 4) dscp (100000)—decimal value 32
cs5	CS5 (precedence 5) dscp (101000)—decimal value 40
cs6	CS6 (precedence 6) dscp (110000)—decimal value 48
cs7	CS7 (precedence 7) dscp (111000)—decimal value 56
default	Default dscp (000000)—decimal value 0
ef	EF dscp (101110)—decimal value 46

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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 dscp** *dscp-list*	Configures the traffic class by matching packets based on the values in the *dscp-list* variable. For a list of DSCP values, see the Standard DSCP Values table.
Step 4	(Optional) switch(config-cmap-qos)# **no match dscp** *dscp-list*	Removes the match from the traffic class. For a list of DSCP values, see the Standard DSCP Values table.

Example

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

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

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

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

Configuring Protocol Classification

You can classify traffic based on the IPv4 Protocol field or the IPv6 Next Header field in the IP header. The following table shows the protocol arguments:

Table 5: Protocol Arguments

Argument	Description
arp	Address Resolution Protocol (ARP)
clns_es	CLNS End Systems
clns_is	CLNS Intermediate System
dhcp	Dynamic Host Configuration (DHCP)
ldp	Label Distribution Protocol (LDP)
netbios	NetBIOS Extended User Interface (NetBEUI)

Procedure

	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	(Optional) switch(config-cmap-qos)# **no match protocol** {**arp** | **clns_es** | **clns_is** | **dhcp** | **ldp** | **netbios**}	Removes the match from the traffic class.

Example

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 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.

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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 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	(Optional) switch(config-cmap-qos)# **no match ip rtp** *port-number*	Removes the match from the traffic class.

Example

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

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

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

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

Configuring ACL Classification

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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	(Optional) switch(config-cmap-qos)# **no match access-group name** *acl-name*	Removes the match from the traffic class.

Example

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

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

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

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

QoS ACL Per-Entry Statistics

Starting with Cisco NX-OS Release 7.2(0)N1(1), for ACLs associated with QoS Policy, statistics are shown per ACE.

Due to the way statistics and policers are attached to the TCAM entries, there are certain limitations to viewing the statistics:

- Statistics per ACE in an ACL cannot be viewed if there is more than one ACE in the ACL and a policer is attached to the QoS policy.
- The above limitation applies to qos-based matches as well (for example, **match dscp** *value*, **match precedence** *value*, and so on).
  - Statistics cannot be viewed with match-all rules.
  - Statistics can be viewed only with match-any.
- Statistics per-ACE of ACL for QoS policies applied of FEX HIF ports will be shown only if policer is not present.

Example: Enabling QoS Policy Statistics

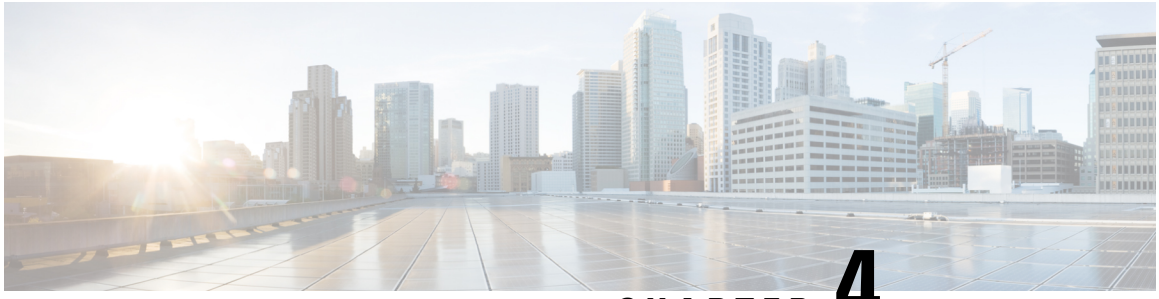
Statistics will be enabled if the user provides statistics per-entry in the ACL, which is used in QoS Policies.

```
Switch(config-acl)# show ip access-lists test_ACL1

IPV4 ACL test_ACL1
        statistics per-entry
       10 permit ip 10.10.10.1/24 20.2.2.2/24 ------>//Operation when a policer is attached//

        20 deny ip 40.4.4.4/24 any
        30 permit ip 30.3.3.3/24 11.11.11.1/24
Switch(config-acl)#
Switch(config-acl)# class-map type qos test_map
Switch(config-cmap-qos)# match access-group name test_ACL1
Switch(config-cmap-qos)# exit
Switch(config)# policy-map type qos test_pmap
Switch(config-pmap-qos)# class test_map
Switch(config-pmap-c-qos)# set qos-group 4
Switch(config-pmap-c-qos)# conf
Switch(config)# int e1/26
Switch(config-if)# service-policy type qos input test_pmap
Switch(config-if)# conf
Switch(config)# show ip access-lists test_ACL1

IPV4 ACL test_ACL1
        statistics per-entry
        10 permit ip 10.10.10.1/24 20.2.2.2/24 [match=0]--->//Operation with no policer
attached or ACL having only one entry//
        20 deny ip 40.4.4.4/24 any [match=0]
        30 permit ip 30.3.3.3/24 11.11.11.1/24 [match=0]
```

Verifying the Classification Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show class-map	Displays the class maps defined on the switch.
show policy-map [*name*]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
running-config ipqos	Displays information about the running configuration for QoS.
startup-config ipqos	Displays information about the startup configuration for QoS.



CHAPTER 4

Configuring Policy Maps

This chapter contains the following sections:

Information About Policy Types

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

There are three policy types:

- Network-qos
- Queuing
- QoS

The following QoS parameters can be specified for each type of class:

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

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

- 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 device associated with system classes.

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

Note

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

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

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

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

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

Configuring Policy Maps

Creating Policy Maps

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

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

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

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

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

Before you begin

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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:

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

Configuring Type 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.

Note If FCoE QoS policy is configured and offloaded to FEX without configuring the FCoE Network QoS policy, offloaded QoS policy on the FEX is unable to identify the FCoE class and therefore, QoS policy will not be applied on the FCoE traffic. Hence it is required to have the FCoE network QoS policy configured before offloading the FCoE QoS policy to the FEX.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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	(Optional) switch(config-pmap-c-nq)# **no mtu**	Resets the MTU value in this class.
Step 6	switch(config-pmap-c-nq)# **pause no-drop**	Configures a no-drop class.
Step 7	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 8	(Optional) switch(config-pmap-c-nq)# **no set cos** *cos-value*	Disables the marking operation in this class.

Example

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

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

Configuring Type QoS Policies

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

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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** | **class-default**] **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.

Example

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

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

Configuring Type 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.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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,

	Command or Action	Purpose
		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)# **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 5	(Optional) switch(config-pmap-c-que)# **no priority**	Removes the strict priority queuing from the traffic in this class.
Step 6	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 7	(Optional) switch(config-pmap-c-que)# **no bandwidth percent** *percentage*	Removes the bandwidth specification from this class.

Example

Enabling and Displaying QoS Statistics for Interfaces

The **qos statistics** command must be enabled when you migrate to Cisco NX-OS release 7.3(0)N1(1) to enable statistics for policy maps. In addition, the existing policies need to be removed and reassigned to ensure the statistics work. Any new policy maps (that are not already configured on any interface) configured after enabling statistics would have the statistics enabled.

From Cisco NX-OS release 7.3(2)N1(1), the following changes are introduced:

- You do not need to use the **qos statistics** command to enable the QoS statistics. By default, the QoS statistics is enabled.
- The class-map and match statistics are not supported. Both these statistics are not displayed when you run the **show policy-map interface** command.
- The policer statistics are still supported and the queuing statistics are added to the **show policy-map interface** command output. Note that the queuing statistics are supported only for the Ethernet interfaces.

The detailed procedure is as follows:

Before you begin

Enabling statistics can take up additional TCAM space. Hence, you must ensure that there is enough space available to perform this operation, given the existing configuration. Refer to CSCuq00149 for details on whether the statistics can be enabled on your switch.

Procedure

Step 1 Enter global configuration mode:

switch# **configure terminal**

Step 2 (Optional) Verify the existing status of the statistics on your switch:

switch(config)# **show policy-map vlan** *vlan-number*

Step 3 (Optional) Enable the statistics, if the existing status of the statistics is disabled on your switch:

switch(config)# **qos statistics**

Note From Cisco NX-OS release 7.3(2)N1(1), the QoS statistics is enabled by default.

Step 4 Enter VLAN configuration mode for the specified VLAN:

switch(config)# **vlan configuration** *vlan-number*

Step 5 Remove the policy from the VLAN:

switch(config-vlan-config)# **no service-policy type qos input** *policy-name*

The *policy-name* is the name assigned to the policy map.

Step 6 Enter VLAN configuration mode for the specified VLAN:

switch(config-vlan-config)# **vlan configuration** *vlan-number*

Step 7 Assign or reapply the policy map to the VLAN:

switch(config-vlan-config)# **service-policy type qos input** *policy-name*

Note The policy-name is the name assigned to the policy map. Note that the policy must be removed from all attachment points (VLANs and interfaces) before you enable the statistics on even one attachment point.

Step 8 Verify the status of the statistics on your switch:

switch(config-vlan-config)# **show policy-map vlan** *vlan-number*

Note You can also use the **show policy-map interface** command.

Note To remove the statistics, use the **no qos statistics** command, then remove and reassign the policies for them to take effect.

Example: Enabling and Displaying QoS Statistics

This example shows how to enable and display QoS statistics.

```
switch(config)# show policy-map vlan 13

Global statistics status: disabled

Vlan 13

  Service-policy (qos) input:   rq1
    policy statistics status:   disabled

    Class-map (qos):   rq1 (match-any)
      Match: cos 4
      set qos-group 2

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

switch(config)# qos statistics

Warning: Turning on the statistics would increase the TCAM utilisation. Disable the CLI if
 this is not intended.
Note that the policies need to be removed and re-applied, for statistics to take effect.


switch(config)#vlan configuration 13-59
switch(config-vlan-config)#no service-policy type qos input rq1
switch(config-vlan-config)#vlan configuration 13
switch(config-vlan-config)#service-policy type qos input rq1
switch(config-vlan-config)# show policy-map vlan 13

Global statistics status: enabled

Vlan 13

  Service-policy (qos) input:   rq1
    policy statistics status:   enabled

    Class-map (qos):   rq1 (match-any)
      3094788 packets
      Match: cos 4
        3094788 Match packets
      set qos-group 2

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

switch(config-vlan-config)#
```

The following example shows the output of the **show policy-map interface** command in the Cisco NX-OS release 7.3(2)N1(1).

```
switch(config)# show policy-map interface ethernet 1/49


Global statistics status :   enabled
```

```
 NOTE: Type qos policy-map configured on VLAN will take precedence
       over system-qos policy-map for traffic on the VLAN


Ethernet1/49

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

    Class-map (qos):   cos3 (match-all)
      Match: cos 3
      set qos-group 2
      police cir percent 60 bc 200 ms
        conformed 300579840 bytes, 899939640 bps action: transmit
        violated 43806000 bytes, 131155688 bps action: drop

    Class-map (qos):   cos4 (match-all)
      Match: cos 4
      set qos-group 4

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

  Service-policy (queuing) input:   fcoe-default-in-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

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

    Class-map (queuing):   class-fcoe (match-any)
      Match: qos-group 1
      queue dropped pkts : 0  queue received pkts : 0
      bandwidth percent 50

    Class-map (queuing):   class-default (match-any)
      Match: qos-group 0
      queue dropped pkts : 57346780  queue received pkts : 155740874
      bandwidth percent 50
```

Restrictions for Viewing the QoS Statistics

Due to the way statistics and policers are attached to the TCAM entries, there are certain limitations to viewing the statistics:

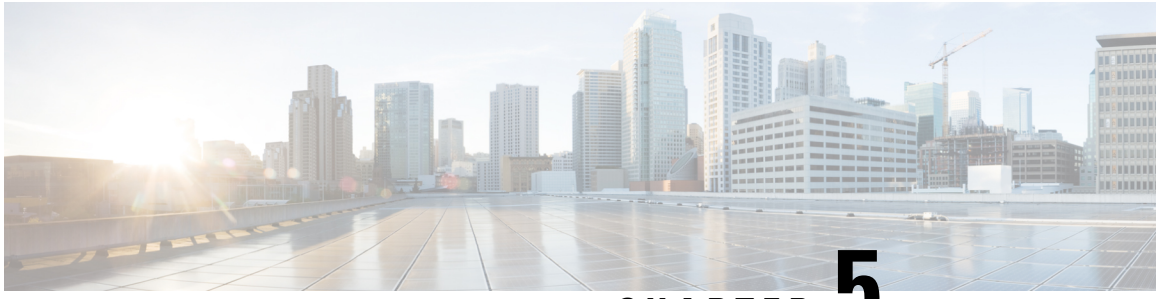
- Statistics are incremented cumulatively for each VLAN or interface where the policy is applied. The statistics are not per-interface/vlan.
- Default policies and system-level policies do not have statistics.
- Statistics per ACE in an ACL cannot be viewed if there is more than one ACE in the ACL and a policer is attached to the QoS policy.
- The above limitation applies to qos-based matches as well (for example, **match dscp** *value*, **match precedence** *value*, and so on).

- Statistics cannot be viewed with match-all rules.
- Statistics can be viewed only with match-any.

- Statistics per-ACE of ACL for QoS policies applied of FEX HIF ports will be shown only if policer is not present.

Verifying the Policy Map Configuration

Command	Purpose
show policy-map [*name*]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
show policy-map interface [*interface number*]	Displays the policy map settings for an interface or all interfaces.
show policy-map system	Displays the policy map settings attached to the system qos.
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.
running-config ipqos	Displays information about the running configuration for QoS.
startup-config ipqos	Displays information about the startup configuration for QoS.



CHAPTER 5

Configuring Marking

This chapter contains the following sections:

Information About Marking

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

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

- DSCP
- IP precedence
- CoS

Configuring Marking

Configuring DSCP Marking

For Cisco Nexus devices, 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 6: Standard DSCP Values

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

Procedure

	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

	Command or Action	Purpose
		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 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 from 1 to 5.

Example

This example 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

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

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

Table 7: 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)

Value	List of Precedence Values
priority	Priority precedence (1)
routine	Routine precedence (0)

Procedure

	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.

Example

```
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

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

Procedure

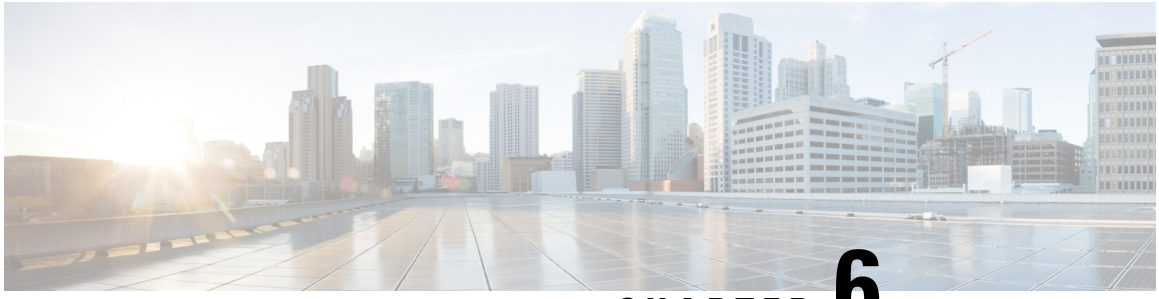
	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **policy-map** [**type network-qos**] *policy-map name*	Creates or accesses the policy map named *policy-map-name* and enters policy-map mode. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.

	Command or Action	Purpose
Step 3	switch(config-pmap-nq) # **class** [**type network-qos**] {*class-map name* |**class-default**}	Creates a reference to the *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	switch(config-pmap-c-nq) # **set cos** *cos-value*	Specifies the CoS value to cos-value. The *cos-value* can range from 0 to 7. **Note** This command is supported only for egress policies.

Verifying the Marking Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show class-map	Displays the class maps defined on the switch.
show policy-map [*name*]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
running-config ipqos	Displays information about the running configuration for QoS.
startup-config ipqos	Displays informationa bout the startup configuration for QoS.



CHAPTER 6

Configuring QoS on the System

This chapter contains the following sections:

Information About System Classes

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 device 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.

Default System Classes

MTU

The Cisco Nexus device 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 2158 bytes. If the Cisco Nexus device 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 2158 bytes and the maximum MTU is 9216 bytes.
- Configuring the MTU to 9216 bytes on both the Layer 3 ports and the Network QoS at the same time is not supported.
- 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 switch sends the MTU configuration to network adapters that support DCBX.

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

Configuring System QoS

Attaching the System Service Policy

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **system qos**	Enters system class configuration mode.

Example

Restoring the Default System Service Policies

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

Procedure

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

	Command or Action	Purpose
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.

Example

Configuring the Queue Limit for a Specified Fabric Extender

At the Fabric Extender configuration level, you can control the queue limit for a specified Fabric Extender for egress direction (from the network to the host). You can use a lower queue limit value on the Fabric Extender to prevent one blocked receiver from affecting traffic that is sent to other noncongested receivers ("head-of-line blocking"). A higher queue limit provides better burst absorption and less head-of-line blocking protection. You can use the **no** form of this command to allow the Fabric Extender to use all available hardware space.

Note At the system level, you can set the queue limit for Fabric Extenders by using the **fex queue-limit** command. However, configuring the queue limit for a specific Fabric Extender will override the queue limit configuration set at the system level for that Fabric Extender.

You can specify the queue limit for the following Fabric Extenders:

- Cisco Nexus 2148T Fabric Extender (48x1G 4x10G SFP+ Module)
- Cisco Nexus 2224TP Fabric Extender (24x1G 2x10G SFP+ Module)
- Cisco Nexus 2232P Fabric Extender (32x10G SFP+ 8x10G SFP+ Module)
- Cisco Nexus 2248T Fabric Extender (48x1G 4x10G SFP+ Module)
- Cisco Nexus N2248TP-E Fabric Extender (48x1G 4x10G Module)
- Cisco Nexus N2348UPQ Fabric Extender (48x10G SFP+ 6x40G QSFP Module)

Procedure

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

	Command or Action	Purpose
Step 2	switch(config)# **fex** *fex-id*	Specifies the Fabric Extender and enters the Fabric Extender mode.
Step 3	switch(config-fex)# **hardware** *fex_card_type* **queue-limit** *queue-limit*	Configures the queue limit for the specified Fabric Extender. The queue limit is specified in bytes. The range is from 81920 to 652800 for a Cisco Nexus 2148T Fabric Extender and from 2560 to 652800 for all other supported Fabric Extenders.

Example

This example shows how to restore the default queue limit on a Cisco Nexus 2248T Fabric Extender:

```
switch# configure terminal
switch(config-if)# fex 101
switch(config-fex)# hardware N2248T queue-limit 327680
```

This example shows how to remove the queue limit that is set by default on a Cisco Nexus 2248T Fabric Extender:

```
switch# configure terminal
switch(config)# fex 101
switch(config-fex)# no hardware N2248T queue-limit 327680
```

Enabling the Jumbo MTU

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

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

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

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

```
switch# conf t
switch(config)# policy-map type network-qos jumbo
switch(config-pmap-nq)# class type network-qos class-fcoe
switch(config-pmap-nq-c)# end
```

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

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

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

Verifying the Jumbo MTU

On the Cisco Nexus device, traffic is classified into one of eight QoS groups. The MTU is configured at the QoS group level. By default, all Ethernet traffic is in QoS group 0. To verify the jumbo MTU for Ethernet traffic, use the **show queueing interface ethernet** *slot/chassis_number* command and find "HW MTU" in the command output to check the MTU for QoS group 0. The value should be 9216.

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

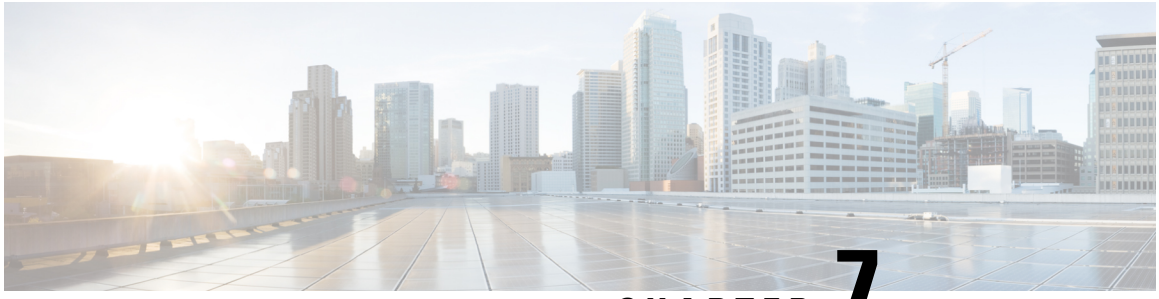
This example shows how to display jumbo MTU information for Ethernet 1/19:

```
switch# show queuing interface ethernet1/19
Ethernet1/19 queuing information:
  TX Queuing
    qos-group  sched-type  oper-bandwidth
        0       WRR             50
        1       WRR             50

  RX Queuing
    qos-group 0
    q-size: 243200, HW MTU: 9280 (9216 configured)
    drop-type: drop, xon: 0, xoff: 1520
    Statistics:
        Pkts received over the port             : 2119963420
        Ucast pkts sent to the cross-bar        : 2115648336
        Mcast pkts sent to the cross-bar        : 4315084
        Ucast pkts received from the cross-bar  : 2592447431
        Pkts sent to the port                   : 2672878113
        Pkts discarded on ingress               : 0
        Per-priority-pause status               : Rx (Inactive), Tx (Inactive)

    qos-group 1
    q-size: 76800, HW MTU: 2240 (2158 configured)
    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
        Per-priority-pause status               : Rx (Inactive), Tx (Inactive)

  Total Multicast crossbar statistics:
    Mcast pkts received from the cross-bar      : 80430744
```

Verifying the System QoS Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show policy-map system	Displays the policy map settings attached to the system QoS.
show policy-map [*name*]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
show class-map	Displays the class maps defined on the switch.
running-config ipqos	Displays information about the running configuration for QoS.
startup-config ipqos	Displays information a bout the startup configuration for QoS.



CHAPTER 7

Configuring QoS on Interfaces

This chapter contains the following sections:

Information About Interface QoS

Trust Boundaries

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

- By default, all Ethernet interfaces are trusted interfaces.The 802.1p CoS and DSCP are preserved unless the marking is configured. There is no default CoS to queue and DSCP to queue mapping. You can define and apply a policy to create these mappings. By default, without a user defined policy, all traffic is assigned to the default queue.
- Any packet that is not tagged with an 802.1p CoS value is classified into the default drop system class. If the untagged packet is sent over a trunk, it is tagged with the default untagged CoS value, which is zero.
- You can override the default untagged CoS value for an Ethernet interface or port channel.

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.

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.

Configuring Interface QoS

Configuring Untagged CoS

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** {**ethernet** [*chassis*/]*slot*/*port* | **port-channel** *channel-number*}	Enters the configuration mode for the specified interface or port channel.
Step 3	switch(config-if)# **untagged cos** *cos-value*	Configures the untagged CoS value. Values can be from 1 to 7.

Example

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

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

Configuring an Interface Service Policy

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** {**ethernet** [*chassis*/]*slot*/*port* | **port-channel** *channel-number*}	Enters the 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 input** *policy-name*	Applies the policy map to the interface.

	Command or Action	Purpose
		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.

Example

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

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

Configuring a Service Policy for a Layer 3 Interface

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

Procedure

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

Example

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

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

The following example shows how to attach an input qos policy map to a Layer 3 interface:

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

Changing the Bandwidth Allocated to Unicast and Multicast Traffic

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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 bandwidth-value percentage ranges from 0 to 100 percent.

Example

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

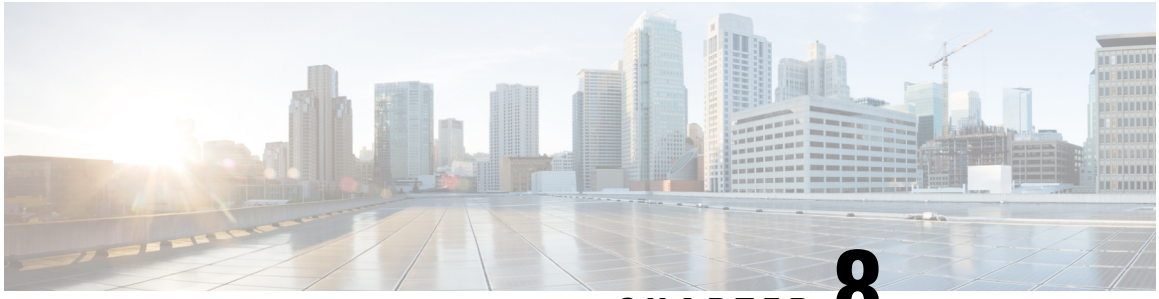
```
switch# configure terminal
switch(config)# interface ethernet 1/5
switch(config-if)# wrr unicast-bandwidth 75
switch(config-if)#
```

Verifying the Interface QoS Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show class-map	Displays the class maps defined on the switch.
show policy-map [*name*]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
show policy-map interface [*interface number*]	Displays the policy map settings for an interface or all interfaces.
show queuing interface *[interface slot/\port*]	Displays the queue configuration and statistics.
show interface flowcontrol [**module** *numbef*]	Displays the detailed listing of the flow control settings on all interfaces.
show interface [*interface slot/port*] **priority-flow-control** [**module** *number*]	Displays the priority flow control details for a specified interface.
show interface untagged-cos [**module** *number*]	Displays the untagged CoS values for all interfaces.

Command	Purpose
running-config ipqos	Displays information about the running configuration for QoS.
startup-config ipqos	Displays information about the startup configuration for QoS.



CHAPTER 8

Configuring QoS on VLANs

This chapter contains the following sections:

Information About VLAN QoS

On Cisco Nexus devices, you can configure quality of service (QoS) policies for classification and marking on VLANs. The policies that you apply to a VLAN are applied to the traffic on the VLAN's Layer 2 and switch virtual interface (SVI) ports.

Precedence of QoS Policies

The marking requirements in a QoS policy determine its precedence. Interface QoS policies take the highest precedence, the VLAN QoS policies are next, and the System QoS policies have the lowest precedence.

However, if a VLAN is assigned both a VLAN QoS policy and a VLAN ACL (VACL), the VACL takes the highest precedence.

Example of Interface, System, and VLAN Policy Precedence

This example shows a configuration where the traffic on interface 1/1 with CoS 5 goes to qos-group 3. Traffic on the other interfaces with VLAN 10 and CoS 5 go to qos-group 4. Traffic on interfaces other than VLAN 10 and CoS 5 go to qos-group 5.

```
class-map type qos match-all cm1
  match cos 5
policy-map type qos pm-ifc
  class cm1
    set qos-group 3
```

```
  class class-default
policy-map type qos pm-vlan
  class cm1
    set qos-group 4
  class class-default
policy-map type qos pm-sys
  class cm1
    set qos-group 5
  class class-default

system qos
  service-policy type qos input pm-sys
vlan configuration 10
  service-policy type qos input pm-vlan
interface Ethernet1/1
  service-policy type qos input pm-ifc
```

Example of Interface and System QoS Policy Precedence

This example shows a configuration where the traffic on interface 1/1 with CoS 5 goes to qos-group 3. Traffic on the other interfaces with CoS 5 go to qos-group 5.

```
class-map type qos match-all cm1
  match cos 5
policy-map type qos pm-ifc
  class cm1
    set qos-group 3
  class class-default
policy-map type qos pm-sys
  class cm1
    set qos-group 5
  class class-default

system qos
  service-policy type qos input pm-sys

interface Ethernet1/1
  service-policy type qos input pm-ifc
```

Example of System and VLAN Policy Precedence

This example shows a configuration where the traffic on VLAN 10 with CoS 5 goes to qos-group 4. Traffic on the other VLANs with CoS 5 go to qos-group 5.

```
class-map type qos match-all cm1
  match cos 5
policy-map type qos pm-vlan
  class cm1
    set qos-group 4
  class class-default
policy-map type qos pm-sys
  class cm1
    set qos-group 5
  class class-default

system qos
  service-policy type qos input pm-sys
vlan configuration 10
  service-policy type qos input pm-vlan
```

Example of VLAN QoS and VACL Policy Precedence

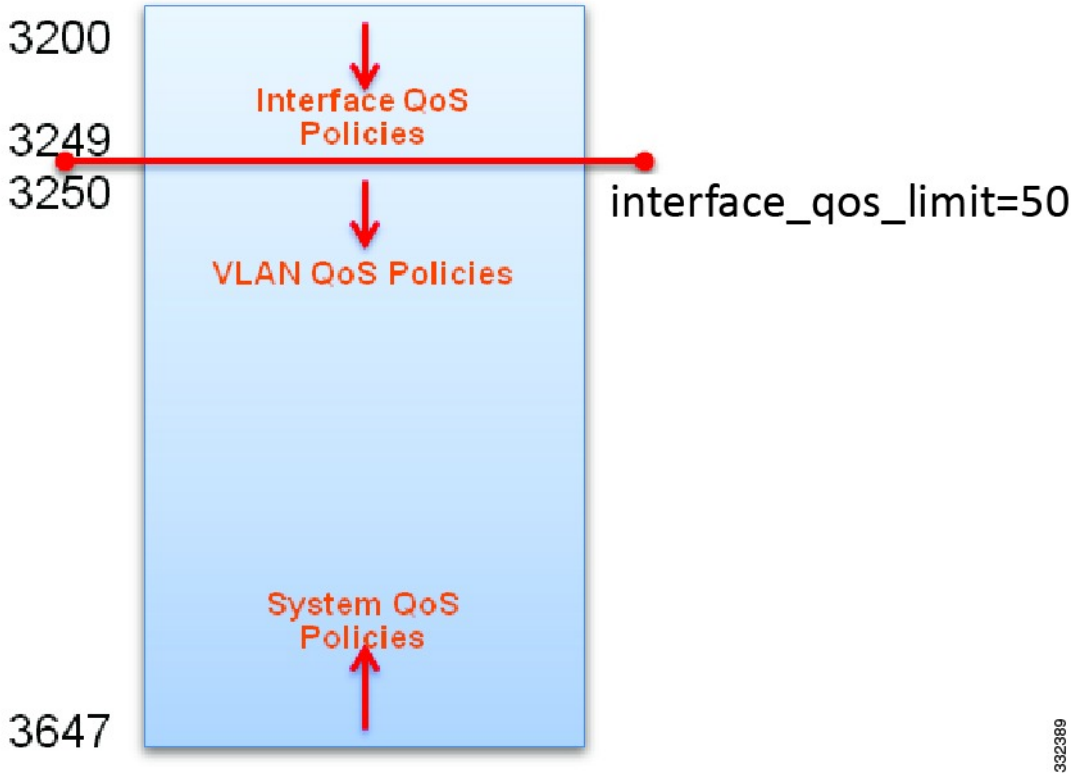
In this example, the packets with source IP address 10.10.10.1 are dropped. However, the other packets with VLAN 10 and CoS 5 go to qos-group 4.

```
ip access-list al1
  10 permit ip 10.10.10.1/24 any
vlan access-map v-am1
  match ip address al1
  action drop
vlan filter v-am1 vlan-list 10

class-map type qos match-all cm1
  match cos 5
policy-map type qos pm-vlan
  class cm1
    set qos-group 4
  class class-default

vlan configuration 10
  service-policy type qos input pm-vlan
```

Limiting TCAM Entries for VLAN QoS

The QoS TCAM region is shared by the interface QoS, system QoS, and VLAN QoS policies. You need to limit the number of TCAM entries for the interface QoS policies in order to define VLAN QoS policies. Use the **hardware profile tcam feature interface-qos limit** *tcam-size* to configure this limit.

Figure 1: QoS TCAM Region



Guidelines and Limitations for VLAN QoS

- A VLAN must have at least one active member port for a service policy to be configured on. If a VLAN does not have at least one active member, and you configure a service policy on it, the configuration is accepted; however, the TCAM is not programmed.
- If a VLAN is removed with the **no vlan** *number* command, the service policy that is configured on that VLAN is still present, but it is not active.
- The TCAM must have enough free entries to configure the service policy on the VLAN.
- A rollback might fail if the interface QoS limit is different in the running configuration than in the rollback configuration.
- If a VLAN with a QoS policy is configured on an interface with no QoS policy, the **show policy-map interface** *number* command does not display the QoS policy configured on the VLAN.
- Remove all interface QoS policies before changing the interface QoS limit.
- Acllogs can only support logging levels of 3 or later.
- We support only logging denials on the ACL, permits will not be logged.
- Only one log message will be displayed until the flow stops and the rest is displayed later.

Configuring VLAN QoS

Configuring or Changing the Interface QoS TCAM Limit

To configure the interface_qos_limit to a specific number, the QoS region of the TCAMs in all of the ASICs cannot have any interfaces policies configured beyond the offset of that number. For example, to configure the interface_qos_limit to 1000, the QoS regions of the TCAMs in all of the ASICs cannot have any interface policies configured beyond offset 1000.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware profile tcam feature interface-qos limit** *tcam-size*	Configures the interface QoS TCAM limit. The *tcam-size* range is from 7 to 446 entries.
Step 3	switch(config)# **show hardware profile tcam feature qos**	Displays the limits of the QoS TCAMs.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to set the interface QoS TCAM limit to 20 entries:

```
switch(config)# configure terminal
switch(config)# hardware profile tcam feature interface-qos limit 20
switch(config)# show hardware profile tcam feature qos
Feature                  Limit (number of tcam entries)
-----------------------------------------------------
interface-qos              20
vlan-qos + global-qos     428

switch(config)# copy running-config startup-config
```

Removing the Interface QoS Limit from the TCAM

Before you begin

- Remove all VLAN QoS policies.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **show hardware profile tcam feature qos**	Displays the limits of the QoS TCAMs.
Step 3	switch(config)# **no hardware profile tcam feature interface-qos limit** *tcam-size*	Configures the interface QoS TCAM limit. The *tcam-size* range is from 7 to 446 entries.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to remove the interface QoS TCAM limit:

```
switch(config)# configure terminal
switch(config)# show hardware profile tcam feature qos
Feature                  Limit (number of tcam entries)
-------------------------------------------------------
interface-qos              20
vlan-qos + global-qos     428

switch(config)# no hardware profile tcam feature interface-qos limit 20
switch(config)# copy running-config startup-config
```

Configuring a Service Policy on a VLAN

Before you begin

- You must configure the interface QoS limit.
- You must configure a policy map.
- The TCAM must have enough free entries to configure the service policy on the VLAN.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **vlan configuration** *vlan-number*	Creates a VLAN and enters VLAN configuration mode. The *vlan-number* range is from 1 to 4094.
Step 3	switch(config-vlan)# **service-policy type qos input** *policy-name*	Assigns a policy map to the VLAN. The *policy-name* is the name assigned to the policy

	Command or Action	Purpose
		map. The name can be a maximum of 40 alphanumeric characters.
Step 4	(Optional) switch(config-vlan)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create a service policy and assign it to VLAN 10:

```
switch# configure terminal
switch(config)# class-map type qos cm1
switch(config-cmap-qos)# match cos 5
switch(config-cmap-qos)# policy-map type qos pm-vlan
switch(config-pmap-qos)# class cm1
switch(config-pmap-c-qos)# set qos-group 4
switch(config-pmap-c-qos)# exit
switch(config-pmap-qos)# exit
switch(config)# vlan configuration 10
switch(config-vlan-config)# service-policy type qos input pm-vlan
switch(config-vlan-config)#
```

Removing a Service Policy from a VLAN

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **vlan configuration** *vlan-number*	Enters VLAN configuration mode for the specified VLAN. The *vlan-number* range is from 1 to 4094.
Step 3	switch(config-vlan-config)#**no service-policy type qos input** *policy-name*	Removes the policy from the VLAN. The *policy-name* is the name assigned to the policy map. The name can be a maximum of 40 alphanumeric characters.
Step 4	(Optional) switch(config-vlan-config)# **copy running-config startup-config**	Saves the changes persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to remove the pm-vlan policy map from VLAN 10:

```
swtich# configure terminal
switch(config)# vlan configuration 10
switch(config-vlan-config)# no service-policy type qos input pm-vlan
switch(config-vlan-config)# copy running-config startup-config
```

Verifying the VLAN QoS Configuration

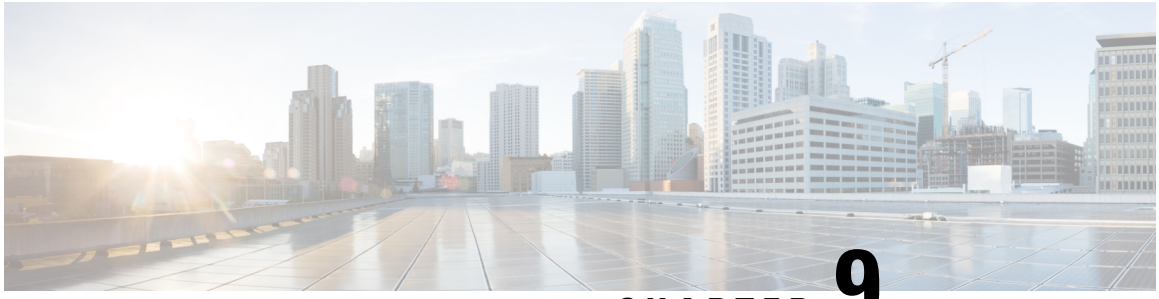
Use one of the following commands to verify the configuration:

Command	Purpose
show policy-map vlan *vlan-number*	Displays the QoS policies configured on the specified VLAN.
show policy-map [*name*]	Displays the policy maps defined on the switch. Optionally, you can display the named policy only.
running-config ipqos	Displays information about the running configuration for QoS.
startup-config ipqos	Displays information about the startup configuration for QoS.

Feature History for VLAN QoS

Table 8: Feature History for VLAN QoS

Feature Name	Release	Feature Information
VLAN QoS	5.1(3)N2(1)	This feature was introduced.



CHAPTER 9

Configuring Queuing and Flow Control

This chapter contains the following sections:

Information About Queues

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.

Egress Queuing Policies

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

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

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

- In addition to these queues, control traffic that is destined for the CPU uses strict priority queues. These queues are not accessible for user configuration.
- Standard Ethernet traffic in the default drop system class is assigned a queue. This queue uses WRR scheduling with 100 percent of the bandwidth.

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

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

Buffering and Queue Limits on the Cisco Nexus 5500 Platform

On the Cisco Nexus device, the packet buffer per port is 640KB.

On the Nexus 5500 platform, the packet buffer per port is 640KB. The Nexus 5548P, Nexus 5548UP, and the Nexus 5596UP switch share the same buffer architecture. The Nexus 5500 platform implements Virtual Output Queueing (VOQ) and ingress buffer architecture with the majority of the buffer allocated at ingress. The architecture allows the switch to store packets at multiple ingress ports when there are multiple ports sending traffic to one egress port which causes congestion.

The following default buffer allocations per port exist for the Cisco Nexus 5500 Platform:

Table 9: Cisco Nexus 5500 Platform Default Buffer Allocations Per Port

Traffic Class	Ingress Buffer (KB)
Class-fcoe	79.360
User-defined no-drop with an MTU less than 2240	79.360
User-defined no-drop class with an MTU greater than 2240	90.204
Tail drop traffic class	22.720
Class-default	All of the remaining buffer (470 with default QoS configuration)

The default buffer allocation varies depending on the type of class. For example, if you create a regular tail drop traffic class the default allocation is 22.7KB, unless you specify a larger size using the **queue-limit** command.

To increase the ingress buffer space available to a user-created qos-group, from a network-qos policy-map, use the **queue-limit** command.

In addition to ingress buffer allocated for each user-created qos-group there is an additional 29.76KB buffer required at egress for each qos-group.

With the default QoS configuration, all of the available buffer (470KB) is allocated to the class-default. When you create a new qos-group, the buffer required for the new qos-group will be taken away from class-default. The amount of buffer that is left for class-default equals 470 minus the ingress buffer used by other qos-groups minus 29.76KB and times the number of qos-groups.

Note Each new class requires an additional 29.76KB, so the exact amount of buffer that is left in the class default equals 470 minus the buffer used by other qos-groups minus 29.76KB times the number of qos-groups.

The default QoS policy for the Cisco Nexus device does not create class-fcoe and does not reserve buffer and qos-group for FCoE traffic.

The **show queuing interface** command can display the amount of ingress buffer allocated for each qos-group

Information About Flow Control

Link-Level Flow Control

IEEE 802.3x link-level flow control allows a congested receiver to communicate a transmitter at the other end of the link to pause its data transmission for a short period of time. The link-level flow control feature applies to all the traffic on the link.

The transmit and receive directions are separately configurable. By default, link-level flow control is disabled for both directions.

On the Cisco Nexus device, Ethernet interfaces do not automatically detect the link-level flow control capability. You must configure the capability explicitly on the Ethernet interfaces.

On each Ethernet interface, the switch can enable either priority flow control or link-level flow control (but not both).

Priority Flow Control

Priority flow control (PFC) allows you to apply pause functionality to specific classes of traffic on a link instead of all the traffic on the link. PFC applies pause functionality based on the IEEE 802.1p CoS value. When the switch enables PFC, it communicates to the adapter which CoS values to apply the pause.

Note You cannot enable PFC on FEX N2K-C2248TP-E-1GE.

Ethernet interfaces use PFC to provide lossless service to no-drop system classes. PFC implements pause frames on a per-class basis and uses the IEEE 802.1p CoS value to identify the classes that require lossless service.

In the switch, each system class has an associated IEEE 802.1p CoS value that is assigned by default or configured on the system class. If you enable PFC, the switch sends the no-drop CoS values to the adapter, which then applies PFC to these CoS values.

The default CoS value for the FCoE system class is 3. This value is configurable.

By default, the switch negotiates to enable the PFC capability. If the negotiation succeeds, PFC is enabled and link-level flow control remains disabled regardless of its configuration settings. If the PFC negotiation fails, you can either force PFC to be enabled on the interface or you can enable IEEE 802.x link-level flow control.

If you do not enable PFC on an interface, you can enable IEEE 802.3X link-level pause.

Note Ensure that pause no-drop is configured on a class map for link-level pause.

By default, link-level pause is disabled.

Configuring Queuing

Configuring the Queue Limit for a Specified Fabric Extender

At the Fabric Extender configuration level, you can control the queue limit for a specified Fabric Extender for egress direction (from the network to the host). You can use a lower queue limit value on the Fabric Extender to prevent one blocked receiver from affecting traffic that is sent to other noncongested receivers ("head-of-line blocking"). A higher queue limit provides better burst absorption and less head-of-line blocking protection. You can use the **no** form of this command to allow the Fabric Extender to use all available hardware space.

Note At the system level, you can set the queue limit for Fabric Extenders by using the **fex queue-limit** command. However, configuring the queue limit for a specific Fabric Extender will override the queue limit configuration set at the system level for that Fabric Extender.

You can specify the queue limit for the following Fabric Extenders:

- Cisco Nexus 2148T Fabric Extender (48x1G 4x10G SFP+ Module)
- Cisco Nexus 2224TP Fabric Extender (24x1G 2x10G SFP+ Module)
- Cisco Nexus 2232P Fabric Extender (32x10G SFP+ 8x10G SFP+ Module)
- Cisco Nexus 2248T Fabric Extender (48x1G 4x10G SFP+ Module)
- Cisco Nexus N2248TP-E Fabric Extender (48x1G 4x10G Module)
- Cisco Nexus N2348UPQ Fabric Extender (48x10G SFP+ 6x40G QSFP Module)

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **fex** *fex-id*	Specifies the Fabric Extender and enters the Fabric Extender mode.
Step 3	switch(config-fex)# **hardware** *fex_card_type* **queue-limit** *queue-limit*	Configures the queue limit for the specified Fabric Extender. The queue limit is specified in bytes. The range is from 81920 to 652800 for a Cisco Nexus 2148T Fabric Extender and from 2560 to 652800 for all other supported Fabric Extenders.

Example

This example shows how to restore the default queue limit on a Cisco Nexus 2248T Fabric Extender:

```
switch# configure terminal
switch(config-if)# fex 101
switch(config-fex)# hardware N2248T queue-limit 327680
```

This example shows how to remove the queue limit that is set by default on a Cisco Nexus 2248T Fabric Extender:

```
switch# configure terminal
switch(config)# fex 101
switch(config-fex)# no hardware N2248T queue-limit 327680
```

Configuring No-Drop Buffer Thresholds

You can configure the no-drop buffer threshold settings for 3000m lossless Ethernet.

Note To achieve lossless Ethernet for both directions, the devices connected to the device must have the similar capability. The default buffer and threshold value for the no-drop can ensure lossless Ethernet for up to 300 meters.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **policy-map type network-qos** *policy-map name*	Enters policy-map network-qos class mode and identifies the policy map assigned to the type network-qos policy map.
Step 3	switch(config-pmap-nq)# **class type network-qos** *class-map name*	References an existing network QoS class map in a policy map and enters class mode.
Step 4	switch(config-pmap-nq-c)# **pause no-drop buffer-size** *buffer-size* **pause-threshold** *xoff-size* **resume-threshold** *xon-size*	Specifies the buffer threshold settings for pause and resume for 3000m lossless Ethernet: • buffer-size—Buffer size for ingress traffic, in bytes. Valid values are from 10240 to 490880. **Note** On a Cisco Nexus 5020 switch, you can configure a maximum buffer size of 143680 bytes. On a Cisco Nexus 5500 Series device, you can configure a maximum buffer size of 152000 bytes. • pause-threshold—Specifies the buffer limit at which the port pauses the peer.

	Command or Action	Purpose
		• xoff-size—Buffer limit for pausing, in bytes. Valid values are 0 to 490880. **Note** On a Cisco Nexus 5020 switch, you can configure a maximum pause-threshold value of 58860 bytes. On a Cisco Nexus 5500 Series device, you can configure a maximum pause-threshold value of 103360 bytes. • resume-threshold—Specifies the buffer limit at which the port resumes the peer. • xon-size—Buffer limit at which to resume, in bytes. Valid values are 0 to 490880. **Note** On a Cisco Nexus 5020 switch, you can configure a maximum resume-threshold value of 38400 bytes. On a Cisco Nexus 5500 Series device, you can configure a maximum resume-threshold value of 83520 bytes.
Step 5	(Optional) switch(config-pmap-nq-c)# **no pause no-drop buffer-size** *buffer-size* **pause-threshold** *xoff-size* **resume-threshold** *xon-size*	Removes the buffer threshold settings for pause and resume for 3000m lossless Ethernet.
Step 6	switch(config-pmap-nq-c)# **exit**	Exits class mode.
Step 7	switch(config-pmap-nq)# **exit**	Exits policy-map network-qos mode.

Example

This example shows how to configure the no-drop buffer threshold for 3000 meters.

```
switch(config-pmap-nq)# policy-map type network-qos nqos_policy
switch(config-pmap-nq)# class type network-qos nqos_class
switch(config-pmap-nq-c)# pause no-drop buffer-size 152000 pause-threshold 103360
resume-threshold 83520
switch(config-pmap-nq-c)# exit
switch(config-pmap-nq)# exit
switch(config)# exit
switch#
```

Configuring the Buffer Threshold for the Cisco Nexus 2148T Fabric Extender

In the Fabric Extender configuration mode, you can configure the buffer threshold for the Cisco Nexus 2148T Fabric Extender. The buffer threshold sets the consumption level of input buffers before an indication is sent to the egress queue to start observing the tail drop threshold. If the buffer usage is lower than the configured buffer threshold, the tail drop threshold is ignored.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **fex** *fex-id*	Specifies the Fabric Extender and enters the Fabric Extender mode.
Step 3	switch(config-fex)# **hardware N2148T buffer-threshold** *buffer limit*	Configures the buffer threshold for the Cisco Nexus 2148T Fabric Extender. The buffer threshold is specified in bytes. The range is from 81920 to 316160 for the Cisco Nexus 2148T Fabric Extender.

Example

This example shows how to restore the default buffer threshold on the Cisco Nexus 2148T Fabric Extender:

```
switch# configure terminal
switch(config)# fex 101
switch(config-fex)# hardware N2148T buffer-threshold 163840
```

This example shows how to remove the default buffer threshold on the Cisco Nexus 2148T Fabric Extender:

```
switch# configure terminal
switch(config)# fex 101
switch(config-fex)# no hardware N2148T buffer-threshold
```

Enabling Virtual Output Queuing Limits for Unicast Traffic on the Cisco Nexus Device

You can enable the Virtual Output Queuing (VOQ) limit for unicast traffic. To alleviate congestion and blocking, use VOQ to prevent one blocked receiver from affecting traffic that is sent to other noncongested blocking receivers.

Procedure

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

	Command or Action	Purpose
Step 2	switch(config)# **hardware unicast voq-limit**	Enables the VOQ limit for unicast traffic. The default is disabled.
Step 3	switch(config)# **no hardware unicast voq-limit**	Disables the VOQ limit for unicast traffic.

Example

This example shows how to enable the VOQ limits for unicast packets on a switch:

```
switch(config)# hardware unicast voq-limit
switch(config)#
```

Configuring Flow Control

Link-Level Flow Control

IEEE 802.3x link-level flow control allows a congested receiver to communicate a transmitter at the other end of the link to pause its data transmission for a short period of time. The link-level flow control feature applies to all the traffic on the link.

The transmit and receive directions are separately configurable. By default, link-level flow control is disabled for both directions.

On the Cisco Nexus device, Ethernet interfaces do not automatically detect the link-level flow control capability. You must configure the capability explicitly on the Ethernet interfaces.

On each Ethernet interface, the switch can enable either priority flow control or link-level flow control (but not both).

Configuring Priority Flow Control

By default, Ethernet interfaces negotiate PFC with the network adapter using the DCBX protocol. When PFC is enabled, PFC is applied to traffic that matches the CoS value configured for the no-drop class.

You can override the negotiation result by forcing the interface to enable PFC.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type slot/port*	Specifies the interface to be changed.
Step 3	switch(config-if)# **priority-flow-control mode** {**auto** | **on**}	Sets PFC mode for the selected interface. Specifies auto to negotiate PFC capability. This is the default.

	Command or Action	Purpose
		Specifies on to force-enable PFC.
Step 4	(Optional) switch(config-if)# **no priority-flow-control mode on**	Disables the PFC setting for the selected interface.

Example

This example shows how to force-enable PFC on an interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# priority-flow-control mode on
```

Configuring Link-Level Flow Control

By default, LLC on Ethernet interfaces is disabled. You can enable LLC for the transmit and receive directions.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type slot/port*	Specifies the interface to be changed.
Step 3	switch(config-if)# **flowcontrol** [**receive** {**on** | **off**}] [**send** {**on** | **off**}]	Enables LLC for the selected interface. Set **receive** and/or **send on** or **off**.
Step 4	(Optional) switch(config-if)# **no flowcontrol** [**receive** {**on** | **off**}] [**send** {**on** | **off**}]	Disables LLC for the selected interface.

Example

This example shows how to enable LLC on an interface:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# interface e1/48
switch(config-if)# flowcontrol receive on
switch(config-if)# flowcontrol send on
```

Verifying the Queue and Flow Control Configurations

Use one of the following commands to verify the configuration:

Command	Purpose
show queuing interface *[interface slot/\port]*	Displays the queue configuration and statistics.

Command	Purpose
show interface flowcontrol [**module** *numbef*]	Displays the detailed listing of the flow control settings on all interfaces.
show interface [*interface slot/port*] **priority-flow-control** [**module** *number*]	Displays the priority flow control details for a specified interface.
show wrr-queue cos-map [*var*]	
running-config ipqos	Displays information about the running configuration for QoS.
startup-config ipqos	Displays informationa bout the startup configuration for QoS.

CHAPTER 10

Configuring Ingress Policing

This chapter contains the following sections:

Information About Ingress Policing

Policing allows you to monitor the data rates for a particular class of traffic. When the data rate exceeds user-configured values, the switch drops packets immediately. Because policing does not buffer the traffic; transmission delays are not affected. When traffic exceeds the data rate on a specific class, the switch drops the packets.

You can define single-rate and two-color Ingress Policing.

Single-rate Ingress Policing monitors the committed information rate (CIR) of traffic.

Note The committed information rate (CIR) is a value specified as a bit rate from 1 to 80000000000 or a percentage of the link rate.

In addition, Ingress Policing can monitor associated burst sizes of the packets. Two colors, or conditions, are determined by Ingress Policing for each packet depending on the data rate parameters that you supply.

You can configure only one action for each condition. For example, you might police for traffic in a class to conform to the data rate of 256000 bits per second with up to 200 millisecond bursts.

Color-aware Ingress Policing assumes that traffic has been previously marked with a color.

Table 10: Maximum Supported Hardware Configuration for Policers

	Nexus 5500 Series	Nexus 2232	Nexus 2248TP-E	Nexus 6000 Series
Burst Size	64 MB	32 MB	32 MB	64 MB

	Nexus 5500 Series	Nexus 2232	Nexus 2248TP-E	Nexus 6000 Series
Max Rate	96 Gbps	12 Gbps	8 Gbps	8 Gbps
Granularity	732 kbps	732 kbps	488 kbps	122 kbps

Guidelines and Limitations for Ingress Policing

- The configuration for Ingress Policing is a part of the Quality of Service (QoS) policy configuration. You can configure QoS policies with Ingress Policing on the following :
  - Layer 2 switch ports
  - Host interface (HIF) ports
  - Port channels with switch ports
  - Port channels with HIF ports
  - Layer 3 interfaces (but not sub-interfaces or Switched Virtual Interfaces (SVIs))
  - Virtual Port Channel (vPC)
- Statistics are provided with Ingress Policing. Statistics include the drop count and allowed count. You can display the statistics by entering the **show policy-map interface ethernet** command.
- QoS policies that you configure on the attachments are installed in the QoS region of the Ternary Content Addressable Memory (TCAM) and causes the switch to apply Ingress Policing.
- If you configure a QoS policy with Ingress Policing on a HIF port or HIF port channel, Ingress Policing is offloaded to the Fabric Extender (FEX). Policy rewrites occur only in the switch. So QOS policy offload to FEX is required if there is any QOS policy rewrites which affects policer.
- All the match/set criteria that are supported in a QoS policy are supported even with Ingress Policing present in the policy. A Fabric Extender (FEX) supports Layer 3 operations (fragments) and Layer 4 operations (source and destination port ranges) but not the Transmission Control Protocol (TCP) flags and Layer 2 operations.
- You can define match criteria for a QoS policy so that it matches the control protocol traffic. If the type of policy is configured with Ingress Policing on an HIF port, the control traffic also gets policed. Therefore, the match criteria must be specific to the required flow of traffic.
- The switch cannot apply a QoS policy with Ingress Policing to an HIF port that has virtual Ethernet interfaces attached.
- If the switch applies Ingress Policing on the HIF port, the policer is applied to traffic with no Virtual Network Tag (VNTAG).
- A policy with Ingress Policing is allowed only on switch ports, HIF ports, and port channels with switch/HIF ports.
- Ingress Policing with Layer 2 operations and TCP flags in the match criteria is not allowed on FEX interfaces.
- Ingress Policing is not supported on Enhanced VPC (2LayerVPC) ports.

- It is recommended that you apply identical Ingress Policing on Dual-homed (AA) HIF interfaces.
- The **police** command is not supported on system QoS policies.
- The **show policy-map interface** command is recommended to check that the ingress rate limiter is conformed and to display violated statistics. The CLI displays conformed/violated packets and packet per second statistics on HIF interfaces (regular as well as port-channel), whereas on the switchport (regular as well as port-channel) the command displays conformed/violated bytes and bits per second (bps).

Creating a Policy Map Using a Committed Information Rate

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **policy-map** [**type qos**] [*qos-policy-map-name*]	Creates a named object that represents a set of policies that are 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-map-name* | **class-default**}	Associates a class map with the policy map and enters configuration mode for the specified system class. Use the **class-default** keyword to select all traffic that is currently not matched by classes in the policy map. The *class-map-name* argument can be a maximum of 40 characters. The name is case sensitive and can only contain alphanumeric characters, hyphens, and underscores.
Step 4	switch(config-pmap-c-qos)# **police** [**cir**] {*committed-rate* [*data-rate*] | **percent** *cir-link-percent*} [[**bc**] {*committed-burst-rate*}][**conform**{**transmit**} [**violate** {**drop**}]]]	Polices **cir** in bits, kbps, mbps, or gbps. The **conform** action is applied if the data rate is less than or equal to **cir**, otherwise, the **violate** action is applied. The **cir** keyword specifies to use the committed information rate, or desired bandwidth, as a bit rate or a percentage of the link rate. The *committed-rate* value can range from 1 to 80 Gbps. The *data-rate* value can be one of the following: • bps—bits per second • kbps—1000 bits per second • mbps—1,000,000 bits per second

	Command or Action	Purpose
		• Gbps—1,000,000,000 bits per second Values for *committed-burst-rate* are the following: • bytes—bytes • kbytes—1000 bytes • mbytes—1,000,000 bytes • ms—milliseconds • us—microseconds The following are the Ingress Policing actions: • **conform**—The action to take if the traffic data rate is within bounds. The default action is **transmit.** • **transmit**—Transmits the packet. This is available only when the packet conforms to the parameters. • **violate**—The action to take if the traffic data rate violates the configured rate values. The basic and default action is **drop.** • **drop**—Drops the packet. This action is available only when the packet exceeds or violates the parameters.
Step 5	switch(config-pmap-c-qos)# **exit**	Exits policy-map class configuration mode and enters policy-map mode.
Step 6	switch(config-pmap-qos)# **exit**	Exits policy-map mode and enters configuration mode.
Step 7	(Optional) switch(config)# **show policy-map** [**type qos**] [*policy-map-name*]	Displays information about all configured policy maps or a selected policy map of **type qos**.

Example

This example shows how to create a policy map with Ingress Policing using the committed information rate:

```
switch# configure terminal
switch(config)# policy-map type qos pm1
switch(config-pmap-qos)# class type qos cm1
switch(config-pmap-c-qos)# police cir 10 mbps bc 20 kbytes
switch(config-pmap-c-qos)# set qos-group 4
```

```
switch(config-pmap-c-qos)# end
switch# show policy-map type qos pm1

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

  policy-map type qos pm1
  class type qos cm1
  set qos-group 4
  police cir 20 mbytes conform transmit violate drop
  set qos-group 4
  class type qos class-default
  set qos-group 1
  switch#
```

Creating a Policy Map Using a Percentage of the Interface Rate

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **policy-map** [**type qos**] [*qos-policy-map-name*]	Creates a named object that represents a set of policies that are 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-map-name* | **class-default**}	Associates a class map with the policy map and enters configuration mode for the specified system class. Use the **class-default** keyword to select all traffic that is currently not matched by classes in the policy map. The *class-map-name* argument can be a maximum of 40 characters. The name is case sensitive and can only contain alphanumeric characters, hyphens, and underscores.
Step 4	switch(config-pmap-c-qos)# **police** [**cir**] {*committed-rate* [*data-rate*] | **percent** *cir-link-percent*} [[**bc**] {*committed-burst-rate*}][**conform**{**transmit**} [**violate** {**drop**}]]]	Polices **cir** in bits, kbps, mbps, or gbps. The **conform** action is applied if the data rate is less than or equal to **cir**, otherwise, the **violate** action is applied. The **cir** keyword specifies to use the committed information rate, or desired bandwidth, as a bit rate or a percentage of the link rate. The *cir-link-percent* value can range from 1 to 100 percent.

	Command or Action	Purpose
		Values for *committed-burst-rate* are the following: • bytes—bytes • kbytes—1000 bytes • mbytes—1,000,000 bytes The following are the Ingress Policing actions: • **conform**—The action to take if the traffic data rate is within bounds. The default action is **transmit.** • **transmit**—Transmits the packet. This is available only when the packet conforms to the parameters. • **violate**—The action to take if the traffic data rate violates the configured rate values. The basic and default action is **drop.** • **drop**—Drops the packet. This is available only when the packet exceeds or violates the parameters.
Step 5	switch(config-pmap-c-qos)# **exit**	Exits policy-map class configuration mode and enters policy-map mode.
Step 6	switch(config-pmap-qos)# **exit**	Exits policy-map mode and enters configuration mode.
Step 7	(Optional) switch(config)# **show policy-map** [**type qos**] [*policy-map-name* | **qos-dynamic**]	Displays information about all configured policy maps or a selected policy map of **type qos**.

Example

This example shows how to create a policy map with Ingress Policing using the percentage of the interface rate:

```
switch# configure terminal
switch(config)# policy-map type qos pm-test1
switch(config-pmap-qos)# class type qos cm-cos4
switch(config-pmap-c-qos)# police cir percent 10 bc 40 kbytes conform transmit violate drop
switch(config-pmap-c-qos)# end
switch# show policy-map type qos pm-test1

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

policy-map type qos pm-test1
```

```
class type qos cm-cos4
set qos-group 4
police cir percent 10 bc 40 kbytes conform transmit violate drop
class type qos class-default
set qos-group 1
switch#
```

Verifying Ingress Policing Configuration

To verify Ingress Policing configuration information, perform one of the following tasks:

Command	Purpose
switch# **show policy-map interface** [*interface number*]	Displays the policy map settings for an interface or all interfaces.
switch# **show policy-map** [**type qos**] [*policy-map-name*]	Displays information about all configured policy maps or a selected policy map of **type qos**.

Configuration Examples for Ingress Policing

The following example shows the Committed Information Rate (CIR) being specified as a percentage where the Ingress Policing rate is calculated based on the port/port-channel speed:

```
switch(config)# policy-map type qos pm-cos
switch(config-pmap-qos)# class cm-cos
switch(config-pmap-c-qos)# police cir percent 10 bc 20 mbytes conform transmit violate drop

switch(config-pmap-c-qos)#
```

The following example shows the output of the **show policy-map** command with Ingress Policing configured:

```
switch(config-pmap-c-qos)# show policy-map pm-cos


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

  policy-map type qos pm-cos
    class type qos cm-cos
      set qos-group 4
      police cir percent 10 bc 20 mbytes conform transmit violate drop
    class type qos class-default
      set qos-group 1
switch(config-pmap-c-qos)#
```

The following example shows a policy being applied to an interface with the **service-policy** command:

```
switch(config)# interface ethernet 1/1
switch(config-if)# service-policy type qos input pm-cos
```

The following example shows policy statistics being displayed by using the **show policy-map** command:

```
switch(config-if)# show policy-map interface ethernet 1/1
Global statistics status :   disabled

Ethernet1/1

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

    Class-map (qos):   qos-police (match-all)
      0 packets
      Match: dscp 10
      police cir percent 100 bc 200 ms
        conformed 0 bytes, 0 bps action: transmit
        violated 0 bytes, 0 bps action: drop
```

CHAPTER 11

Egress Multicast Buffer

Effective with Cisco NX-OS Release 7.2(0)N1(1), the Cisco Nexus 5600 and 6000 series switches support egress multicast buffer tuning.

Information About Egress Multicast Buffering

The egress multicast buffer allows additional cells to enhance multicast traffic at egress. When there is heavy multicast traffic, buffer space (cells) is borrowed from the unicast pool. The pool provides a specific number of cells to enhance the burst absorption and minimize traffic drops at egress.

Note The egress multicast buffering feature is enabled only in multicast heavy traffic environment. Also, there is a set limit for the multicast pool size (the number of cells that can be borrowed).

You can configure multicast traffic buffering when the traffic patten in your environment has:

- Multicast traffic counts 90 to 99 percent
- Multicast traffic is inconsistent

Configuring Egress Multicast Buffer Tuning

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware multicast-buffer-tune**	Tunes the egress multicast buffer size to have more burst absorption.

Example

The following example shows how the egress multicast buffer size is tuned to have more burst absorption.

```
switch# configure terminal
switch(config)# hardware multicast-buffer-tune
```

Verifying Egress Multicast Buffering

To verify the egress multicast buffering, use the following command:

Command	Purpose
switch# **show platform software qd info**	Displays the status of the multicast buffer enable feature, the allocated space for egress buffer, and number of cells in unicast and multicast pool in 40G and 10G fabric modes.

The following example shows how to verify cells values in unicast and multicast pool before enabling the egress multicast buffer command.

```
Switch# show platform software qd info

Multicast buffer enable feature is : Disabled //verifying unicast
and multicast cell values when multicast
buffer feature is disabled//

Egress buffer allocation
Fabric mode : 40G Fabric Mode
      10G Port                        40G Port
pool| total| xoff|  xon| xcos| cls|   total| xoff|  xon| xcos| cls
uc 0|   100|   60|   30|    0|  00|     100|   60|   30|    0|  00
uc 1|     0|    0|    0|    0|  00|       0|    0|    0|    0|  00
uc 2|     0|    0|    0|    0|  00|       0|    0|    0|    0|  00
uc 3|  1035|  700|   16|  350|  fe|    1934| 1512|   64| 1112|  fe
mc 0| 13292|    0|    0|  -  |  ff|   20666|    0|    0|  -  |  ff
mc 1|     0|    0|    0|  -  |  00|       0|    0|    0|  -  |  00
```

The following example shows how to verify cell values in unicast and multicast pool after enabling the egress multicast buffer command:

```
Switch# show platform software qd info

Multicast buffer enable feature is : Enabled //verifying unicast and
multicast cell values when multicast
buffer feature is enabled//

Egress buffer allocation
Fabric mode : 40G Fabric Mode
      10G Port                        40G Port
pool| total| xoff|  xon| xcos| cls|   total| xoff|  xon| xcos| cls
uc 0|   100|   60|   30|    0|  00|     100|   60|   30|    0|  00
uc 1|     0|    0|    0|    0|  00|       0|    0|    0|    0|  00
```

```
uc 2|     0|    0|    0|    0|  00|       0|    0|    0|    0|  00
uc 3|   451|  116|   16|   87|  fe|     586|  164|   64|  123|  fe
mc 0| 20300|    0|    0|  -  |  ff|   24710|    0|    0|  -  |  ff
mc 1|     0|    0|    0|  -  |  00|       0|    0|    0|  -  |  00
```

CHAPTER 12

Micro-Burst Monitoring Overview

The micro-burst monitoring feature allows you to monitor traffic on a per-port basis for both ingress and egress ports and to detect unexpected data bursts within a very small time window (micro-seconds). This allows you to detect flows in the network that are at risk of data loss, and that may require extra bandwidth.

A micro-burst occurs when a specific amount of data (in bytes) is exceeded in a given time interval. The micro-burst monitoring feature allows you to specify these limits as absolute values (for data and burst size) or as a percentage of the link speed. When these thresholds are exceeded the system generates a Syslog alarm message.

• Micro-Burst Monitoring, on page 77

Micro-Burst Monitoring

Information About Micro-Burst Monitoring

How to Use Micro-Burst Monitoring

The micro-burst monitoring feature monitors bursts in real time. The monitoring process also provides an overview of data path issues, and is helpful in identifying potential capacity issues in a network. Syslog messages are generated with the burst exceeds the configured value.

Micro-burst monitoring provides real-time burst information that is used to:

- monitor network micro bursts
- trigger to congestion detection and latency processes

Guidelines and Limitations for Micro-Burst Monitoring

- Micro-burst detection is performed on a per-link basis and port channels are not be taken into consideration.
- Micro-burst detection is supported on Ethernet ports only, and is not supported on Fabric Extender Technology (FEX), Port Channels, Virtual Ethernet (VETH), or Virtual Fibre Channel (VFC) ports.

How to Configure Micro-Burst Monitoring

Configuring Micro-Burst Monitoring

To configure micro-burst monitoring, you first set micro-burst threshold values for an interface and then configure the maximum number of micro-bursts allowed on the interface. You configure ingress and egress port settings separately.

Procedure

	Command or Action	Purpose

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

Step 2	**interface ethernet** *slot/port* **Example:** `switch(config)# interface ethernet 1/1`	Enters interface configuration mode.
Step 3	**burst threshold ingress limit** *percent* **interval** *interval_time* **Example:** `switch(config-if)# burst threshold ingress limit 60 interval 10000000`	Configures micro-burst threshold values for ingress traffic on the interface.
Step 4	**burst threshold egress size** *max_bytes* **interval** *interval_time* **Example:** `switch(config-if)# burst threshold egress size 500000 interval 16000`	Configures micro-burst threshold values for egress traffic on the interface.
Step 5	**burst maximum egress burst-count** *max_bytes* **Example:** `switch(config-if)# burst maximum egress burst-count 50000`	Configures the maximum number of micro-bursts allowed within a time interval before generating an interrupt on a port in the egress direction. This time interval is equal to 10 multiplied by the micro-burst threshold interval (in seconds).
Step 6	**burst maximum ingress burst-count** *max_bytes* **Example:** `switch(config-if)# burst maximum ingress burst-count 600000`	Configures the maximum number of micro-bursts allowed within a time interval before generating an interrupt on a port in the ingress direction. This time interval is equal to 10 multiplied by the micro-burst threshold interval (in seconds).
Step 7	**exit** **Example:** `switch(config-if)# exit`	Updates the configuration and exits interface configuration mode.

	Command or Action	Purpose
Step 8	**clear burst-counters [interface {all | ethernet interface}] {both | egrees | ingress }** **Example:** `switch# clear burst-counters interface all`	Clears the micro-burst counters on all interfaces or only on ethernet interfaces. Additionally, the command is applicable to clear counters for both egress and ingress or either egress or ingress traffic.
Step 9	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Verifying Micro-Burst Monitoring

To display micro-burst monitoring information, enter the following show command:

Command	Purpose
show interface burst-counters	Displays micro-burst counters information for all interfaces where micro-burst monitoring is configured.

Example for Micro-Burst Monitoring

Configuration Example for Micro-Burst Monitoring

The following example shows how to configure micro-burst monitoring on an Ethernet interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/1
switch(config-if)# burst threshold egress limit 50 interval 30
switch(config-if)# burst threshold ingress size 500000 interval 16000
switch(config-if)# burst maximum egress burst-count 50000
switch(config-if)# burst maximum ingress burst-count 600000
switch(config-if)# exit
switch(config)# copy running-config startup-config
```

CHAPTER 13

Switch Latency Monitoring Overview

The switch latency monitoring feature marks each ingress and egress packet with a timestamp value. To calculate the latency for each packet in the system the switch compares the ingress with the egress timestamp. The feature allows you to display historical latency averages between all pairs of ports, as well as real-time latency data.

You can use the latency measurements to identify which flows are impacted by latency issues. In addition the statistics generated by the switch latency monitoring feature allow you to plan network topologies, manage incident responses and identify root causes for application issues in the network. You can also use the statistics to provide a Service Level Agreement (SLA) for latency intensive applications.

Information About Switch Latency Monitoring

How to Use Switch Latency Monitoring

Switch Latency Monitoring feature measure packet latency in nanoseconds. It provides information in the following modes:

- Real time mode maintains minimum, maximum, and average latency values for all packets between input and output port pairs.
- Historical mode maintains flow-based latency distribution histograms and provides linear, exponential, or custom binning.

Switch Latency Monitoring Guidelines and Limitations

Switch Latency Monitoring has the following limitations and guidelines:

- Only one mode (instantaneous, linear histogram, exponential histogram, or custom histogram) can be configured between a egress-ingress port pair at a time. Instantaneous Mode is enabled by default.
- If any histogram mode is configured between a pair of ports then instantaneous mode is disabled.
- If the histogram mode is removed between a pair of ports then instantaneous mode is enabled.

- All switch Latency histogram statistics are lost if the base value is modified.
- When the latency monitoring mode between an ingress and egress port pair is changed, switch latency statistics between that port pair are lost.
- Switch Latency Monitoring records are not maintained across a switch reload or ISSU.
- The Switch Latency Monitoring feature is supported on Ethernet interfaces only.
- You must issue the **clear hardware profile latency monitor all** command when the switch is reloaded, or when a new module is powered on.

Switch Latency Monitoring Modes

Switch latency monitoring is supported in the following four modes:

- Instantaneous Mode

  This mode is enabled by default and allows you to collect minimum, maximum, and average latency values for all the packets flowing between the ingress and egress ports .

- Linear Histogram

  This mode counts how many packets are in a given latency range by allowing you to count packets within ranges of latencies (in nanoseconds) separately. For example, you can configure a linear histogram that counts how many packets fall into each of the following latency ranges: 800-848, 848-896, 896-944, and 944-992. To configure a linear histogram monitoring mode, you first specify a base for the table (in this example 800 nanoseconds) and you then specify a step value (in this example 50 nanoseconds).

- Exponential Histogram

  This mode allows the binning of latencies for ranges that increase in an exponential manner. For example, to count the packets in the following ranges of latencies: 848-896, 896-992, 992-1184, and 1184-1568 you specify the mode as exponential mode and set the base value to 800 nanoseconds and the step to 50 nanoseconds.

- Custom Histogram

  This mode allows you to count the number of packets falling within a specified range, and the number of packets that falls outside of the specified range.

How to Configure Switch Latency Monitoring

Configuring Switch Latency Monitoring

To configure switch latency monitoring you first set the monitoring base value and then configure the ingress and egress port pair and monitoring mode.

Note Switch latency monitoring in instantaneous mode is enabled by default.

Procedure

	Command or Action	Purpose
Step 1	enable **Example:** `switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**clear hardware profile latency monitor all** **Example:** `switch# clear hardware profile latency monitor all`	Clears the statistics for all egress and ingress port pairs in the system.

Step 3	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 4	**hardware profile latency monitor base** *nanoseconds* **Example:** `switch(config)# hardware profile latency monitor base 800`	Specifies the base value used to construct switch latency monitoring histograms. Valid values are multiples of 8 in the range 8 to 2147483640 nanoseconds.
Step 5	**interface ethernet** *slot/port* **Example:** `switch(config)# interface ethernet 1/1`	Enters interface configuration mode. This interface is the egress interface for the egress and ingress port pair.
Step 6	**packet latency interface ethernet** *ingress-interface-slot/port* **mode linear step** *step-value* **Example:** `switch(config-if)# packet latency ethernet 1/2 mode linear step 40`	Configures linear mode monitoring between the egress Ethernet interface and this specified ingress Ethernet interface.
Step 7	**packet latency interface ethernet** *ingress-interface-slot/port* **mode exponential step** *step-value* **Example:** `switch(config-if)# packet latency ethernet 1/3-4 mode exponential step 40`	Configures exponential mode monitoring between the egress Ethernet interface and the specified ingress Ethernet interface ports.
Step 8	**packet latency interface ethernet** *ingress-interface-slot/port* **mode customer low-latency** *low-value* **high-latency** *high-value* **Example:** `switch(config-if)# packet latency ethernet 1/5 mode customer low-latency 40 high-latency 1200`	Configures custom mode monitoring between the egress Ethernet interface and the specified ingress Ethernet interface.

	Command or Action	Purpose
Step 9	**exit** **Example:** `switch(config-if)# exit`	Updates the configuration and exits interface configuration mode.
Step 10	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Verifying Switch Latency Monitoring Statistics

To display the Switch Latency Monitoring statistics, perform the following task:

Command	Purpose
show hardware profile latency monitor interface ethernet *egress-interface-slot/port* **interface ethernet** *ingress-interface-slot/port*	Displays switch latency statistics for specified egress-ingress port pair. **Note** The ASIC reports a high latency value when the number of packets for a given latency range is monitored using the histograms or when the maximum latency value is monitored in the instantaneous mode. The maximum latency value is approximately 0.4 sec, which is incorrect. Ignore the value displayed because this is an unexpected behavior and does not have any functional impact on the switch.

Configuration Examples for Switch Latency Monitoring

Configuration Example for Switch Latency Monitoring

This example shows how to configure switch latency monitoring:

```
switch(config)# hardware profile latency monitor base 800
switch(config)# interface ethernet 1/1
switch(config-if)# packet latency interface ethernet 1/2 mode linear step 40
switch(config-if)# packet latency interface ethernet 1/3-4 mode exponential step 40
switch(config-if)# packet latency interface ethernet 1/5 mode custom low 40 high 1200
switch(config)# interface ethernet 2/1
switch(config-if)# packet latency interface ethernet 1/1 mode exponential step 80
```

CHAPTER 14

WRED-Explicit Congestion Notification Feature Overview

Currently, the congestion control and avoidance algorithms for Transmission Control Protocol (TCP) are based on the idea that packet loss is an appropriate indication of congestion on networks transmitting data using the best-effort service model. When a network uses the best-effort service model, the network delivers data if it can, without any assurance of reliability, delay bounds, or throughput. However, these algorithms and the best-effort service model are not suited to applications that are sensitive to delay or packet loss (for instance, interactive traffic including Telnet, web-browsing, and transfer of audio and video data). Weighted Random Early Detection (WRED), and by extension, Explicit Congestion Notification (ECN), helps to solve this problem.

RFC 3168, *The Addition of Explicit Congestion Notification (ECN) to IP*, states that with the addition of active queue management (for example, WRED) to the Internet infrastructure, routers are no longer limited to packet loss as an indication of congestion.

WRED Explicit Congestion Notification

Information About WRED Explicit Congestion Notification

Guidelines and Limitations for WRED Explicit Congestion Notification

- Explicit Congestion Notification (ECN) parameters are configurable only at system level.
- Weighted Random Early Detection (WRED) cannot be configured alone on a Quality of Service (QoS) group. ECN is enabled by default.
- You must configure WRED thresholds for 10 G interfaces and 40 G interfaces even when no interfaces are present.
- WRED ECN is not applicable to multicast or broadcast traffic.
- WRED ECN is not supported on Nexus 5000 series switches.

How WRED Works

WRED makes early detection of congestion possible and provides a means for handling multiple classes of traffic. WRED can selectively discard lower priority traffic when the router begins to experience congestion and provide differentiated performance characteristics for different classes of service. It also protects against global synchronization. Global synchronization occurs as waves of congestion crest, only to be followed by periods of time during which the transmission link is not used to capacity. For these reasons, WRED is useful on any output interface or router where congestion is expected to occur.

WRED is implemented at the core routers of a network. Edge routers assign IP precedences to packets as the packets enter the network. With WRED, core routers then use these precedences to determine how to treat different types of traffic. WRED provides separate thresholds and weights for different IP precedences, enabling the network to provide different qualities of service, in regard to packet dropping, for different types of traffic. Standard traffic may be dropped more frequently than premium traffic during periods of congestion.

For more information about WRED, refer to the "Congestion Avoidance Overview" module.

ECN Extends WRED Functionality

WRED drops packets, based on the average queue length exceeding a specific threshold value, to indicate congestion. ECN is an extension to WRED in that ECN marks packets instead of dropping them when the average queue length exceeds a specific threshold value. When configured with the WRED -- Explicit Congestion Notification feature, routers and end hosts would use this marking as a signal that the network is congested and slow down sending packets.

As stated in RFC 3168, *The Addition of Explicit Congestion Notification (ECN) to IP,*implementing ECN requires an ECN-specific field that has two bits--the ECN-capable Transport (ECT) bit and the CE (Congestion Experienced) bit--in the IP header. The ECT bit and the CE bit can be used to make four ECN field combinations of 00 to 11. The first number is the ECT bit and the second number is the CE bit. The table below lists each of the ECT and CE bit combination settings in the ECN field and what the combinations indicate.

Table 11: ECN Bit Setting

ECT Bit	CE Bit	Combination Indicates
0	0	Not ECN-capable
0	1	Endpoints of the transport protocol are ECN-capable
1	0	Endpoints of the transport protocol are ECN-capable
1	1	Congestion experienced

The ECN field combination 00 indicates that a packet is not using ECN.

The ECN field combinations 01 and 10--called ECT(1) and ECT(0), respectively--are set by the data sender to indicate that the endpoints of the transport protocol are ECN-capable. Routers treat these two field combinations identically. Data senders can use either one or both of these two combinations. For more information about these two field combinations, and the implications of using one over the other, refer to RFC 3168, *The Addition of Explicit Congestion Notification (ECN) to IP.*

The ECN field combination 11 indicates congestion to the endpoints. Packets arriving a full queue of a router will be dropped.

How Packets Are Treated When ECN Is Enabled

If the number of packets in the queue is below the minimum threshold, packets are transmitted. This happens whether or not ECN is enabled, and this treatment is identical to the treatment a packet receives when WRED only is being used on the network.

If the number of packets in the queue is between the minimum threshold and the maximum threshold, one of the following three scenarios can occur:

- If the ECN field on the packet indicates that the endpoints are ECN-capable (that is, the ECT bit is set to 1 and the CE bit is set to 0, or the ECT bit is set to 0 and the CE bit is set to 1)--and the WRED algorithm determines that the packet should have been dropped based on the drop probability--the ECT and CE bits for the packet are changed to 1, and the packet is transmitted. This happens because ECN is enabled and the packet gets marked instead of dropped.
- If the ECN field on the packet indicates that neither endpoint is ECN-capable (that is, the ECT bit is set to 0 and the CE bit is set to 0), the packet may be dropped based on the WRED drop probability. This is the identical treatment that a packet receives when WRED is enabled without ECN configured on the router.
- If the ECN field on the packet indicates that the network is experiencing congestion (that is, both the ECT bit and the CE bit are set to 1), the packet is transmitted. No further marking is required.

If the number of packets in the queue is above the maximum threshold, packets are dropped based on the drop probability. This is the identical treatment a packet receives when WRED is enabled without ECN configured on the router.

Proxy Queue Drain Rates

When the proxy queue reaches a threshold that indicates congestion, Explicit Congestion Notification (ECN) marking is performed so that the receiver of the packet echoes the congestion indication to the sender. The sender must respond as though the congestion had been indicated by packet drops. The proxy queue drain rate is configured to ensure that during congestion at egress ports only a certain amount of packets are drained. For example, on a 10 Gigabot port, you can configure a drain rate of 9900 Mbps ensuring that not all packets are allowed to drain.

Recommended ECN Thresholds and Proxy Queue Drain Rates

The following table describes the recommended proxy-ques drain rates and maximum and minimum Explicit Congestion Notification (ECN) threshold values.

Parameter	10 Gigabit Port	40 Gigabit Port
ECN min-threshold	64000 bytes	4000 bytes
ECN max-threshold	128000 bytes	256000 bytes
Proxy-queue drain rate	9900 Mbps	39900 Mbps

How to Configure WRED Explicit Congestion notification

Configuring WRED-Explicit Congestion Notification

To configure WRED-ECN you specify interface thresholds, enable ECN, and specify the proxy queue drain rate.

Before you begin

Before you configure Weighted Random Early Detection (WRED) Explicit Congestion Notification (ECN) on the device, you must configure a Quality of Service (QoS) group. In addition, the following restrictions apply:

- Explicit Congestion Notification (ECN) parameters are configurable only at system level.
- Weighted Random Early Detection (WRED) cannot be configured alone on a Quality of Service (QoS) group. ECN is enabled by default.
- You must configure WRED thresholds for 10 G interfaces and 40 G interfaces even when no interfaces are present.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware random-detect min-thresh 10g** *10g-min-threshold* **40g** *40g-min-threshold* **max-thresh 10g** *10g-max-threshold* **40g** *40g-max-threshold* **ecn qos-group** *qos-group-number*	Configures the minimum and maximum thresholds for 10 Gigabit and 40 Gigabit interfaces and enables ECN on particular QoS group. The thresholds for both 10 G and 40 G interfaces can range from 1 to 67108863 bytes. The QoS group number specifies which QoS group is being configured, the range is from 0 (class default) to 5.
Step 3	switch(config)# **hardware pq-drain 10g** *10g-drain-rate* **40g** *40g-drain-rate*	Configures the proxy queue drain rate for 10 Gigabit and 40 Gigabit ports. When congestion occurs at egress ports the drain rate values specify the maximum amount of packets that can be drained. The drain rate for 10 G interfaces can range from 1 to 20000 Mbps. The drain rate for 40 G interfaces can range from 1 to 80000 Mbps.

Example for WRED Explicit Congestion Notification

Configuration Example for WRED Explicit Congestion Notification

The following example shows how to configure Weighted Random Early Detection (WRED) Explicit Congestion Notification (ECN):

```
switch# configuration terminal
switch(config)# hardware random-detect min-thresh 10g 64000 40g 4000 max-thresh 10g 128000
 40g 256000 ecn qos-group 2
switch(config)# hardware pq-drain 10g 9900 40g 39900
switch(config)# exit
switch(config)# copy running-config startup-config
```

CHAPTER 15

Configuring ACL Logging

This chapter contains the following sections:

Information About ACL Logging

The ACL logging feature allows you to monitor ACL flows and to log dropped packets on an interface.

IPv6 ACL Logging Overview

When the ACL logging feature is configured, the system monitors ACL flows and logs dropped packets and statistics for each flow that matches the deny conditions of the ACL entry.

Statistics and dropped-packet logs are generated for each flow. A flow is defined by the source interface, protocol, source IP address, source port, destination IP address, and destination port values. The statistics maintained for a matching flow is the number of denies of the flow by the ACL entry during the specified time interval.

When a new flow is denied (that is a flow that is not already active in the system), the system generates an initial Syslog message with a hit count value of 1. Then each time the flow is denied, the system creates a flow entry and increments the hit count value.

When an existing flow is denied, the system generates a Syslog message at the end of each interval to report the hit count value for the flow in the current interval. After the Syslog message is generated, the hit count value for the flow is reset to zero for the next interval. If no hit is recorded during the interval, the flow is deleted and no Syslog message is generated.

Guidelines and Limitations for ACL Logging

ACL Logging has the following configuration guidelines and limitations:

- The system logs packets that match deny ACE conditions only. Logging for permit ACE conditions is not supported.
- The logging option may be applied to any ACL deny entry. To apply the logging option to implicitly denied traffic, you must configure the logging option for a specific deny-all ACL entry.
- ACL logging applies to port ACLs (PACL) configured by the **ipv6 port traffic-filter** command and to routed ACLs (RACL) configured by the **ipv6 traffic-filter** commands only.
- The total number of flows and deny-flows are limited to a user-defined maximum value to prevent DOS attacks. If this limit is reached, no new logs are created until an existing flow finishes.
- The system uses a hash table to locate a flow so that a large number of flows can be supported without impacting CPU utilization. The system uses a timer queue to efficiently manage the aging of large number of flows.
- The number of Syslog entries generated by the ACL logging process is limited by the configured logging level of the ACL logging process. If the amount of Syslog entries exceed this limit, the logging facility may drop some logging messages. Therefore, ACL logging should not be used as a billing tool or as an accurate source of the number of matches to an access list.
- The hardware rate-limiter rate-limits traffic on a packet basis, but control plane policing (COPP) rate-limits traffic on a byte basis. If the packet size and the hardware rate-limiter both have high values, the COPP default value can be exceeded and the system drops the packet. To overcome this limitation you must increase the default CIR value (64000 bytes) to a higher value such as 2560000 bytes. When the default CIR is increased packet logging happens normally.
- IPv6 logging is not supported on management or VTY (Terminal) ports
- IPv6 logging is not supported on egress VACLs (due to ASIC limitations).

Configuring ACL Logging

To configure the ACL logging process, you first create the access list, then enable filtering of IPv6 traffic on an interface using the specified ACL, and finally configure the ACL logging process parameters.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**ipv6 access-list** *name* **Example:** `switch(config)# ipv6 access-list logging-test`	Defines an IPv6 access list and enters IPv6 access list configuration mode.
Step 3	**deny ipv6 any** *destination-address* **log** **Example:** `switch(config-ipv6-acl)# deny ipv6 any 2001:DB8:1::1/64 log`	Sets deny conditions for an IPv6 access list. To enable the system to log matches against this entry, you must use the **log** keyword when configuring the deny conditions.

	Command or Action	Purpose
Step 4	**exit** **Example:** `switch(config-ipv6-acl)# exit`	Updates the configuration and exits IPv6 access list configuration mode.
Step 5	**interface ethernet** *slot/port* **Example:** `switch(config)# interface ethernet 1/1`	Enters interface configuration mode.
Step 6	**ipv6 traffic-filter logging-test** {**in** | **out**} **Example:** `switch(config-if)# ipv6 traffic-filter logging-test in`	Enables the filtering of IPv6 traffic on an interface using the specified ACL. You can apply an ACL to outbound or inbound traffic.
Step 7	**exit** **Example:** `switch(config-if)# exit`	Updates the configuration and exits interface configuration mode.
Step 8	**logging ip access-list cache interval** *interval* **Example:** `switch(config)# logging ip access-list cache interval 5`	Configures the log-update interval (in seconds) for the ACL logging process. The default value is 300 seconds. The range is from 5 to 86400 seconds.
Step 9	**logging ip access-list cache entries** *number-of-flows* **Example:** `switch(config)# logging ip access-list cache entries 1000`	Specifies the maximum number of flows to be monitored by the ACL logging process. The default value is 8000. The range of values supported is from 0 to 1048576.
Step 10	**logging ip access-list cache threshold** *threshold* **Example:** `switch(config)# logging ip access-list cache threshold 1`	If the specified number of packets are logged before before the expiry of the alert interval the system generates a Syslog message.
Step 11	**hardware rate-limiter access-list-log** *packets* **Example:** `switch(config)# hardware rate-limiter access-list-log 200`	Configures rate limits in packets per second for packets copied to the supervisor module for access list logging. The range is from 0 to 30000.
Step 12	**acllog match-log-level** *severity-level* **Example:** `switch(config)# acllog match-log-level 3`	Specifies the minimum severity level to log ACL matches. The default is 6 (informational). The range is from 0 (emergency) to 7 (debugging). **Note** Acllogs can only support logging levels of 3 or later.

Verifying ACL Logging Configuration

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

Command	Purpose
show logging ip access-list status	Displays the deny maximum flow count, the current effective log interval and the current effective threshold value.
show logging ip access-list cache	Displays information on the active logged flows, such as source IP and destination IP addresses, S-Port and D-Port information and so on.

Configuration Examples for ACL Logging

This example shows how to configure the ACL logging process.

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ipv6 access-list logging-test
switch(config-ipv6-acl)# deny ipv6 any 2001:DB8:1::1/64 log
switch(config-ipv6-acl)# exit
switch(config)# interface ethernet 1/1
switch(config-if)# ipv6 traffic-filter logging-test in
switch(config-if)# exit
switch(config)# logging ip access-list cache entries 1000
switch(config)# logging ip access-list cache interval 5
switch(config)# logging ip access-list cache threshold 1
switch(config)# hardware rate-limiter access-list-log 200
switch(config)# acllog match-log-level 3
switch(config)# exit
switch#
```

This example shows a typical PACL logging configuration.

```
switch(config)# interface ethernet 8/11
switch(config-if)# ipv6 port traffic-filter v6log-pacl in
switch(config-if)# switchport access vlan 4064
switch(config-if)# speed 1000

switch(config)# interface Vlan 4064
switch(config-if)# no shutdown
switch(config-if)# no ip redirects
switch(config-if)# ipv6 address 4064::1/64


Switch# show vlan filter
vlan map v6-vaclmap:
Configured on VLANs: 4064

Switch# show vlan access-map v6-vaclmap
Vlan access-map v6-vaclmap
match ipv6: v6-vacl
```

```
action: drop
statistics per-entry
```

CHAPTER 16

Configuring Buffer Utilization Histogram

This chapter contains the following sections:

Information About the Buffer Utilization Histogram Feature

The Buffer Utilization Histogram feature enables you to analyze the maximum queue depths and buffer utilization in the system in real time. Instantaneous or real time buffer utilization information is supported by the hardware. You can use software to obtain the history of the buffer usage by polling the hardware at regular intervals. Obtaining an historic timeline of the buffer usage provides a better picture of the traffic pattern in the system and helps in traffic engineering. Ultimately, you are able to make better use of the hardware buffer resources.

On the Cisco Nexus device, every three ports of 40 Gigabit Ethernet or every 12 ports of 10 Gigabit Ethernet have access to a shared 25 Mb packet buffer. 15.6 Mb are reserved for ingress and 8.6 Mb are reserved for egress. The remaining space is used for SPAN and control packets.

The Buffer Utilization Histogram enables you to do the following:

- Configure buffer utilization history measurements on the interested ports.
- View buffer utilization over an interval of time.
- Configure either a slow or a fast polling mode.
- Copy collected statistics to the `buffer_util_stats` file on the bootflash drive every hour to allow for later analysis. The collected statistics are appended to the end of the file after an hour and a timestamp is placed in the header that has the interface name.

Guidelines and Limitations for Buffer Utilization Histogram

Buffer Utilization Histogram has the following configuration guidelines and limitations:

- The data is not maintained across upgrades. The new statistics learning restarts after the switch comes online with the new release.
- Unicast and multicast buffer usage can be found in the egress direction. In the ingress direction, buffer usage is combined.
- This feature is supported only on the physical ports. This feature is not supported on virtual interfaces, sub interfaces, FEX Host Interface (HIF) ports, and port channels. Fabric Extender (FEX) fabric ports and port channel member ports are supported.
- You can obtain XML output by using the **show hardware profile buffer monitor** {**all** | **interface** *intf*} **history** {**brief** | **detail**} | **xml** > *filename***.xml** command.

  The previous command displays the XML file content on the CLI. You can redirect it to an XML file. The file can be fetched into any XML analyzer tool for further analysis. It is important to note that XML support is not available for the real time buffer usage. That is, using the command without the **history** option.

Fast Polling

By default, the software is polling the buffer usage every second. Fast polling allows for polling the buffer usage every 250 milliseconds. Changing the polling mode from slow (default value) to fast does not clear the older histogram records that were obtained when you used slow polling mode. When you use the fast polling mode, the older data traverses to the end of the table as the new data keeps coming in. The same scenario applies in the reverse case when the polling mode changes from fast to slow. Although the polling interval is 250 milliseconds for the fast polling mode, the CPU utilization was not affected.

Fast polling results in more granular data. Once the polling mode changes, the polling mode is applied to all the ports on which the Buffer Utilization Histogram feature is enabled.

Default Settings for Buffer Utilization Histogram

The following table lists the default setting for Buffer Utilization Histogram parameters:

Parameters	Default
Buffer Utilization Histogram	Disabled
Polling Mode	Slow

Configuring Buffer Utilization Histogram

Enabling Buffer Utilization Histogram

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** [**ethernet** [*chassis*/]*slot*/*port*]	Enters the configuration mode for the specified interface.
Step 3	switch(config-if)# **hardware profile buffer monitor**	Enables the Buffer Utilization Histogram collection of statistics on a port.
Step 4	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable the Buffer Utilization Histogram Collection feature:

```
switch# configure terminal
switch(config)# interface ethernet 1/1
switch(config-if)# hardware profile buffer monitor
```

Configuring Fast Polling

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware profile buffer monitor sampling fast**	Configures fast polling at an interval of 250 milliseconds.
Step 3	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure fast polling:

```
switch# configure terminal
switch(config)# hardware profile buffer monitor sampling fast
```

Configuring Slow Polling

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no hardware profile buffer monitor sampling fast**	Configures slow polling at an interval of one second.
Step 3	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure slow polling:

```
switch# configure terminal
switch(confg)# no hardware profile buffer monitor sampling fast
```

Disabling the Buffer Utilization Histogram Feature

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** [**ethernet** [*chassis*/]*slot*/*port*]	Enters the configuration mode for the specified interface.
Step 3	switch(config-if)# **no hardware profile buffer monitor**	Disables the Buffer Utilization Histogram collection of statistics on a port.
Step 4	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to disable the Buffer Utilization Histogram feature:

```
switch# configure terminal
switch)config)# interface ethernet 1/1
switch(config-if)# no hardware profile buffer monitor
```

Clearing the Buffer Utilization Histogram History

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **clear hardware profile buffer monitor** [**interface** *ifid*]	Clears the Buffer Utilization Histogram information on a port or all ports based on the parameters supplied. Entering the command without an interface clears the buffer utilization statistics on all ports.

Example

This example shows how to clear the Buffer Utilization Histogram history:

```
switch# configure terminal
swtich(config)# clear hardware profile buffer monitor
```

Verifying the Buffer Utilization Histogram Feature

Use one of the following commands to verify the configuration:

Command	Purpose
show hardware profile buffer monitor {**all** | **interface** *intf*}	Displays the buffer usage statistics for all the unicast and multicast queues on each port and the total global free and used buffer statistics. This command is used to retrieve buffer usage statistics on the basis of instantaneous time (current time).
show hardware profile buffer monitor {**all** | **interface** *intf*} **history** {**brief** | **detail**}	Displays the buffer usage historical statistics for all the ports or a specified port. This command supports brief and detailed representations. The brief representation is used to display only average usages over time, but the detailed representation displays the maximum, minimum, and average usages over time.

Output Examples for Buffer Utilization Histogram

This example shows the output when the polling mode is set to slow. Buffer utilization data is obtained every second. No minimum/maximum/average available for the data in the 1 sec column. Five samples from the 1 sec column make the first entry of the 5 sec column (min/max/avg are calculated from the five samples present in the 1 sec column). Twelve samples from the 5 sec column make the first entry in the 1 min column. Five samples from the 1 min column make the first entry in the 5 min column. Twelve samples from the 5 min

column make the first entry in the 1 hour column. This information is copied to the file on the bootflash. Data propagates in the table in a circular fashion.

```
switch(config)# show hardware profile buffer monitor interface ethernet 1/1 history detail
--------------------------------------------------------------------------------
                                        Interface : Eth1/1
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
                                  Sampling Mode : Slow (1 second)
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
           Ingress Buffer Utilization Detected(Min|Max|Avg)(in KB)
                 Per asic Ingress Total Usage (15.628800MB)
--------------------------------------------------------------------------------
     1 sec     |     5 sec     |     1 min     |     5 min     |     1 hour    |
--------------------------------------------------------------------------------
16.3|  -  | -  |12.5|18.9| 14.9| 9.3|22.7| 15.7| 0.0|23.0| 13.7|      N/A      |
21.4|  -  | -  |13.4|22.7| 17.5| 0.0|22.1|  5.8| 6.7|23.0| 16.3|      N/A      |
12.5|  -  | -  |10.2|21.4| 15.0| 0.0| 0.0|  0.0| 9.3|23.0| 15.8|      N/A      |
13.8|† -  | -  | 9.9|22.1| 13.0| 0.0|22.7|  5.5|      N/A      |      N/A      |
12.8|† -  | -  |10.2|15.4| 12.4| 9.3|23.0| 15.7|      N/A      |      N/A      |
      N/A      |10.9|20.5| 17.4|      N/A      |      N/A      |      N/A      |
      N/A      | 9.3|22.1| 18.0|      N/A      |      N/A      |      N/A      |
      N/A      |14.7|22.4| 17.7|      N/A      |      N/A      |      N/A      |
      N/A      | 9.9|21.1| 16.5|      N/A      |      N/A      |      N/A      |
      N/A      |11.2|20.8| 15.9|      N/A      |      N/A      |      N/A      |
      N/A      | 9.9|18.2| 14.7|      N/A      |      N/A      |      N/A      |
      N/A      |10.2|22.7| 16.1|      N/A      |      N/A      |      N/A      |
--------------------------------------------------------------------------------
         Egress Unicast Buffer Utilization Detected(Min|Max|Avg)(in KB)
                     Per asic Egress Total Usage (8.611850MB)
--------------------------------------------------------------------------------
     1 sec     |     5 sec     |     1 min     |     5 min     |     1 hour    |
--------------------------------------------------------------------------------
0.0|  -  | -  | 0.0|19.8|† 7.9| 0.0|19.8| 13.0| 0.0|19.8| 10.6|      N/A      |
1.0|  -  | -  | 0.0|19.8| 11.9| 0.0|19.8|  0.4| 0.0|19.8| 12.2|      N/A      |
0.0|  -  | -  | 0.0|19.8| 15.9| 0.0| 0.0|  0.0| 0.0|19.8| 11.9|      N/A      |
19.8|  -  | -  | 0.0|19.8| 15.9| 0.0|19.8| 4.0|      N/A      |      N/A      |
0.0|  -  | -  |19.8|19.8| 19.8| 0.0|19.8| 13.0|      N/A      |      N/A      |
      N/A      | 0.0|19.8| 11.9|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0|19.8| 15.9|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0|19.8| 11.9|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0|19.8|  7.9|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0|19.8| 15.9|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0|19.8|  8.6|      N/A      |      N/A      |      N/A      |
      N/A      |19.8|19.8| 19.8|      N/A      |      N/A      |      N/A      |
--------------------------------------------------------------------------------
         Egress Multicast Buffer Utilization Detected(Min|Max|Avg)(in KB)
                     Per asic Egress Total Usage (8.611850MB)
--------------------------------------------------------------------------------
     1 sec     |     5 sec     |     1 min     |     5 min     |     1 hour    |
--------------------------------------------------------------------------------
0.0|  -  | -  | 0.0| 0.0|  0.0| 0.0| 0.0|  0.0| 0.0| 0.0|  0.0|      N/A      |
0.0|  -  | -  | 0.0| 0.0|  0.0| 0.0| 0.0|  0.0| 0.0| 0.0|  0.0|      N/A      |
0.0|  -  | -  | 0.0| 0.0|  0.0| 0.0| 0.0|  0.0| 0.0| 0.0|  0.0|      N/A      |
0.0|  -  | -  | 0.0| 0.0|  0.0| 0.0| 0.0|  0.0|      N/A      |      N/A      |
0.0|  -  | -  | 0.0| 0.0|  0.0| 0.0| 0.0|  0.0|      N/A      |      N/A      |
      N/A      | 0.0| 0.0|  0.0|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0| 0.0|  0.0|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0| 0.0|  0.0|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0| 0.0|  0.0|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0| 0.0|  0.0|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0| 0.0|  0.0|      N/A      |      N/A      |      N/A      |
      N/A      | 0.0| 0.0|  0.0|      N/A      |      N/A      |      N/A      |
```

```
--------------------------------------------------------------------------------
```

This example contains only the average buffer usage values for the appropriate timelines. This output is only the first row of the detail output.

```
switch# show hardware profile buffer monitor interface e1/1 history brief
--------------------------------------------------------------------------------
                              Interface : Eth1/1
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
                       Sampling Mode : Slow (1 second)
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
                      Ingress Buffer Utilization Detected(in KB)
                      Per asic Ingress Total Usage (15.628800MB)
--------------------------------------------------------------------------------
     1 sec     |     5 sec     |     1 min     |     5 min     |     1 hour    |
--------------------------------------------------------------------------------
            0.0|            0.0|            0.0|            0.0|            0.0|
--------------------------------------------------------------------------------
        Egress Buffer Utilization Detected(Unicast|Multicast)(in KB)
                     Per asic Egress Total Usage (8.611850MB)
--------------------------------------------------------------------------------
     1 sec     |     5 sec     |     1 min     |     5 min     |     1 hour    |
--------------------------------------------------------------------------------
    0.0|    0.0|    0.0|    0.0|    0.0|    0.0|    0.0|    0.0|    0.0|    0.0|
--------------------------------------------------------------------------------
```

CHAPTER 17

Configuring FEX-Based ACL Classification

This chapter contains the following sections:

Information About FEX-based ACL Classification

The Fabric Extender (FEX) based Access Control List (ACL) Classification feature uses ternary content addressable memory (TCAM) resources on a FEX to perform ACL-based packet classification of incoming packets on the switch.

Overview of FEX-based ACL Classification

The FEX-based ACL Classification feature uses TCAM resources on a FEX to perform ACL-based packet classification of incoming packets on the switch. When QoS policies are processed on a FEX, the policies are enforced on the switch and on the associated FEX or FEXes.

By default this feature is disabled. When the feature is enabled, and if the existing system-level QoS policy is accepted by the FEX, the QoS policy is enforced by the FEX. If the existing system-level QoS policy is not accepted by the FEX, an error message is displayed and the fabric ports associated with the FEX are error-disabled, which prevents the FEX from being online.

If the feature is disabled, the existing system-level QoS policy is removed from the FEX and the enforcement of the existing QoS policy is changed from ACL-based to CoS-based. The TCAM entries are removed and packet classification on the FEX is done using the cos2q map in the FEX hardware.

When this feature is enabled, QoS policies are enforced as follows:

- System level QoS policies are enforced on a FEX in the ACL-based approach. That is, TCAM entries are created and programed on FEX ASICs. If the QoS policy is not accepted on a FEX, the command is rejected and an error message is generated. A system level QoS policy is always programmed and enforced completely on the switches and all associated FEXes.

- Interface level QoS policies are enforced on the FEX. That is, TCAM entries in the corresponding FEX ASIC are taken and programmed. If the QoS policy is not accepted on the target interface, the command is rejected and an error message is generated.

Guidelines and Limitations for FEX-Based ACL Classification

When you are configuring Fabric Extender (FEX)-based Access Control List (ACL) classification, you should be aware of the following guidelines and limitations:

- FEX-based ACL classification can be configured for the following interfaces:
  - Global
  - Host interface (HIF) ports
  - HIF port channels
  - VPC
  - 2-Layer VPC
- Only QoS policies are applied at system-level, HIF ports and HIF port channels will be offloaded to FEX platforms.
- In each switching subsystem (SS) on the FEX ASIC, interface-level policies are programmed in TCAM entries in a top-down fashion and system-level policies are programmed in a bottom-up fashion.
- All the match and set criteria supported in a QoS policy are supported even when a policer is present in the policy. FEX supports Layer 3 operations (fragments) and Layer 4 operations (source and destination port ranges). However, policies with TCP flags or Layer 2 operations are not allowed on FEX interfaces.
- QoS policies are not supported on HIF ports that have Virtual Ethernet Interfaces (VETHs) attached.
- If a QoS policy is applied to a HIF port, the classification is applied only to incoming traffic with no VNTAG.
- You could define match criteria for a QoS policy so that the criteria also matches the control protocol traffic. If you configure the policy on a HIF port, the control traffic could also get policed. Therefore, the match criteria should be very specific to the required flow of traffic.
- If a QoS policy is configured on a HIF port or a port channel, the policy is enforced by the FEX and not the switch. However, policy rewrites occur on the switch only.
- Because TCAM entries are not available at network interface (NIF) ports, network-to-host (N2H) traffic is not classified in an ACL-based manner. Instead, N2H traffic is classified in a CoS-based manner.
- ACL-based QoS policy offload is supported on the following platforms:
  - N2224TP Fabric Extender 24x1G 2x10G SFP+ Module
  - N2232P Fabric Extender 32x10G SFP+ 8x10G SFP+ Module
  - N2232TM Fabric Extender 32x10GBase-T 8x10G SFP+ Module
  - N2248T Fabric Extender 48x1G 4x10G SFP+ Module
  - N2248TP E Fabric Extender 48x1G 4x10G SFP+ Module

- N2248PQ Fabric Extender 48x10G SFP+ 16x10G SFP+ Module
- N2232TM-E Fabric Extender 32x10GBase-T 8x10G SFP+ Module
- NB22IBM Fabric Extender 14x10G SFP+ 8x10G SFP+ Module
- N2348UPQ Fabric Extender 6x40G QSFP 48x10G SFP+ FEX

- When a policy is offloaded, the number of access control entries (ACEs) in the policy, which are applied on the FEX, should not exceed 30.

Note In Cisco NX-OS Release 7.3(x), the FEX offload capability using interface QoS policies is upto 100 ACEs, and upto only 30 ACEs using system QoS policies.

- ACL-based QoS policy offload is not supported on the following platforms:
  - N2148T Fabric Extender 48x1G 4x10G SFP+ Module

Configuring FEX-Based ACL Classification

Configuring FEX ACL-based QoS Policy Enforcement

To configure the FEX ACL-based QoS policy enforcement, you must enable policy offloading on each Fabric Extender individually. When you enable the feature on a FEX and if the existing system-level QoS policy is accepted by the FEX, the QoS policy is enforced by the FEX. However, if the existing system-level QoS policy is not accepted by the FEX, the fabric ports associated with the FEX are error-disabled, which then prevents the FEX from being online.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**FEX** *chassis_ID* **Example:** `switch(config)# fex 101`	Enters fabric extender configuration mode.
Step 3	**hardware** *card-type* **qos-policy-offload** **Example:** `switch(config-fex)# hardware N2232P qos-policy-offload`	Enables QoS policy offloading on a Cisco Nexus N2232P Fabric Extender. **Note** When a policy is offloaded, the number of access control entries (ACEs) in the policy, which are applied on the FEX, should not exceed 30.

	Command or Action	Purpose
Step 4	**exit** **Example:** `switch(config-fex)# exit`	Updates the configuration and exits fabric extender configuration mode.
Step 5	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Configuring the FEX ACL-based Interface-Level QoS Policy

When FEX ACL-based QoS policy enforcement is enabled and the interface-level QoS policy is applied successfully, two TCAM entries are created at the top of the TCAM region on the FEX ASIC.

Before you begin

You must enable FEX ACL-based QoS policy enforcement on the switch and on any fabric extenders that you want to use.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**class-map type qos match-all** *class-name* **Example:** `switch(config)# class-map type qos match-all cmap-qos01`	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	**match access-group name** *acl-name* **Example:** `switch(config-cmap-qos)# match access-group name acl-01`	Specifies the named ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to this class.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit`	Updates the configuration and exits class map configuration mode.
Step 5	**policy-map type qos** *policy-name* **Example:** `switch(config)# policy-map type qos pmap-qos01`	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.

	Command or Action	Purpose
Step 6	**class** *class-name* **Example:** `switch(config-pmap-qos)# class cmap-qos01`	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 7	**set qos-group** *qos-group-value* **Example:** `switch(config-pmap-c-qos)# set qos-group 1`	Configures one or more **qos-group** values to match on for classification of traffic into this class map. There is no default value.
Step 8	**interface ethernet** *fex-id/slot/port* **Example:** `switch(config)# interface ethernet 127/1/1`	Enters interface configuration mode.
Step 9	**service-policy type qos input** *policy-map-name* **Example:** `switch(config-if)# service-policy type qos input pmap-qos01`	Attaches the policy map to the interface.
Step 10	**exit** **Example:** `switch(config-if)# exit`	Updates the configuration and exits interface configuration mode.

When the policy is successfully applied, two TCAM entries are created at the top of the TCAM region on the FEX ASIC. The following is an example of that TCAM entry:

```
K-keyType, L-label, B-bindcheck, DH-L2DA, CT-cdceTrnst
    L(IF-ifacl V-vacl Q-qos R-rbacl)

    [8]> K:IP (3/0) IN v4 L-[Q-ff/8 ]
    [8]  SA:ffffff00/c0a80200 DA:00000000/00000000
    [8]-> cos_rw:0 cos:4 de_rw:0 de:0 fwd_to_cpu:0 cos2q_ow:1

    [9]> K:IP/ETH (2/0) IN L-[Q-ff/8 ]
    [9]-> cos_rw:0 cos:2 de_rw:0 de:0 fwd_to_cpu:0 cos2q_ow:1
```

Configuring FEX ACL-based System-Level QoS Policy

When FEX ACL-based QoS policy enforcement is enabled and the system-level QoS policy is applied successfully, two TCAM entries are created at the bottom of the TCAM region on the FEX ASIC.

Before you begin

You must enable FEX ACL-based QoS policy enforcement on the switch and on any fabric extenders that you want to use.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**class-map type qos match-all** *class-name* **Example:** `switch(config)# class-map type qos match-all cmap-qos01`	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	**match access-group name** *acl-name* **Example:** `switch(config-cmap-qos)# match access-group name acl-01`	Specifies the named ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to this class.
Step 4	**exit** **Example:** `switch(config-cmap-qos)# exit`	Updates the configuration and exits class map configuration mode.
Step 5	**policy-map type qos** *policy-name* **Example:** `switch(config)# policy-map type qos pmap-qos01`	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 6	**class** *class-name* **Example:** `switch(config-pmap-qos)# class cmap-qos01`	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 7	**set qos-group** *qos-group-value* **Example:** `switch(config-pmap-c-qos)# set qos-group 1`	Configures one or more **qos-group** values to match on for classification of traffic into this class map. There is no default value.
Step 8	**system** *system-name* **Example:** `switch(config)# system qos`	Enters QoS system configuration mode.
Step 9	**service-policy type qos input** *policy-map-name*	Attaches the classification policy to the system.

	Command or Action	Purpose
	Example: `switch(config-sys-qos)# service-policy type qos input pmap-qos01`	
Step 10	**exit** **Example:** `switch(config-sys-qos)# exit`	Updates the configuration and exits QoS system configuration mode.

When the policy is successfully applied, two TCAM entries are created at the bottom of the TCAM region on the FEX ASIC. The following is an example of that TCAM entry:

```
K-keyType, L-label, B-bindcheck, DH-L2DA, CT-cdceTrnst
    L(IF-ifacl V-vacl Q-qos R-rbacl)

    [253]> K:IP (3/0) IN v4 L-[]
    [253]  SA:ffffff00/c0a80200 DA:00000000/00000000
    [253]-> cos_rw:0 cos:4 de_rw:0 de:0 fwd_to_cpu:0 cos2q_ow:1

    [254]> K:ALL (0/0) IN L-[]
    [254]-> cos_rw:0 cos:2 de_rw:0 de:0 fwd_to_cpu:0 cos2q_ow:1
```

Disabling FEX ACL-based QoS Policy Enforcement

You can disable FEX ACL-based QoS policy enforcement for an individual FEX. If you disable the feature the existing system-level QoS policy is removed from the FEX and the enforcement of the existing QoS policy is changed from ACL-based to CoS-based. In addition, the TCAM entries are removed and packet classification on the FEX is done using the cos2q map in the FEX hardware.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**FEX** *chassis_ID* **Example:** `switch(config)# fex 101`	Enters fabric extender configuration mode.
Step 3	**no hardware** *card-type* **qos-policy-offload** **Example:** `switch(config-fex)# no hardware N2232P qos-policy-offload`	Disables QoS policy offloading on a Cisco Nexus N2232P Fabric Extender.
Step 4	**exit** **Example:** `switch(config-fex)# exit`	Updates the configuration and exits fabric extender configuration mode.
Step 5	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Verifying the FEX-Based ACL Classification

To verify FEX-based ACL classification, perform one of these tasks:

Command	Purpose
show running-config	Displays the contents of the currently running configuration file, including information on FEX-based ACL classification settings.
show queueing interface	Displays the queuing information for FEX Ethernet interfaces, including information about the QoS configuration.

Configuration Examples for FEX-based ACL Classification

The following example shows how to create an IPv4 access control list (ACL):

```
switch# configure terminal
switch(config)# ip access-list acl-01
switch(config-acl)# permit ip 192.168.2.0/24 any
switch(config-acl)# statistics
```

The following example shows how to enable the FEX-based ACL Classification feature on the switch and on a Cisco Nexus N2232P Fabric Extender associated with the switch:

```
switch# configure terminal
switch(config)# fex 101
switch(config-fex)# hardware N2232P qos-policy-offload
switch(config-fex)# exit
```

The following example shows how to configure an ACL-based QoS policy at interface-level for use with the FEX ACL-based QoS policy enforcement feature:

```
switch# configure terminal
switch(config)# class-map type qos match-all cmap-qos01
switch(config-cmap-qos)# match access-group name acl-01
switch(config-cmap-qos)# exit
switch(config)# policy-map type qos pmap-qos01
switch(config-pmap-qos)# class cmap-qos01
switch(config-pmap-c-qos)# set qos-group 2
switch(config)# interface ethernet 101/1/1
switch(config-if)# service-policy type qos input pmap-qos01
switch(config-if)# exit
```

The following example shows how to configure an ACL-based QoS policy at system-level for use with the FEX ACL-based QoS policy enforcement feature:

```
switch# configure terminal
switch(config)# class-map type qos match-all cmap-qos01
switch(config-cmap-qos)# match access-group name acl-01
switch(config-cmap-qos)# exit
switch(config)# policy-map type qos pmap-qos01
switch(config-pmap-qos)# class cmap-qos01
switch(config-pmap-c-qos)# set qos-group 2
switch(config)# system qos
switch(config-sys-qos)# service-policy type qos input pmap-qos01
switch(config-sys-qos)# exit
```

The following example shows how to disable the FEX-based ACL Classification feature on the switch and on a Cisco Nexus N2232P Fabric Extender associated with the switch:

```
switch# configure terminal
switch(config)# fex 101
switch(config-fex)# no hardware N2232P qos-policy-offload
switch(config-fex)# exit
```

The following example shows how to display the ACL-based QoS policy configuration:

```
switch(config-pmap-nq)# show queuing interface ethernet 108/1/48
if_slot 40, ifidx 0x1f6b0bc0
Ethernet108/1/48 queuing information:
  Input buffer allocation:
  Qos-group: 0  2  (shared)
  frh: 2
  drop-type: drop
  cos: 0 1 2 3 4 5 6
  xon       xoff      buffer-size
  ---------+---------+-----------
  34560     39680     48640

  Queueing:
  queue    qos-group    cos            priority  bandwidth mtu
  --------+------------+--------------+---------+---------+----
  2        0            0 1 2 3 4 5 6   WRR        10      1600
  4        2                            WRR         0      1600

  Queue limit: 66560 bytes

  Queue Statistics:
  queue  rx              tx
  ------+---------------+---------------
  2      0               5103082
  4      5103093         0

  Port Statistics:
  rx drop         rx mcast drop   rx error        tx drop         mux ovflow
  ---------------+---------------+---------------+---------------+--------------
  0               0               0               0                InActive

  Priority-flow-control enabled: no
  Flow-control status:
  cos     qos-group   rx pause  tx pause  masked rx pause
  -------+-----------+---------+---------+---------------
  0              0    xon       xon       xon
  1              0    xon       xon       xon
  2              0    xon       xon       xon
  3              0    xon       xon       xon
  4              0    xon       xon       xon
  5              0    xon       xon       xon
  6              0    xon       xon       xon
  7            n/a    xon       xon       xon
```

CHAPTER 18

QoS Configuration Examples

This chapter contains the following sections:

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.

Procedure

	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`

	Command or Action	Purpose

		`(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.

Procedure

	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`

	Command or Action	Purpose

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`

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`

	Command or Action	Purpose

		`(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.

Procedure

	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`

INDEX

F

G

I

J

L

M

O

P

Q

R

S

T

U

V

W