

AOS-CX 10.17 Quality of Service Guide

8400 Switch Series



**Hewlett Packard
Enterprise**

Published: November 2025

Version: 1

Copyright Information

© Copyright 2025 Hewlett Packard Enterprise Development LP.

This product includes code licensed under certain open source licenses which require source compliance. The corresponding source for these components is available upon request. This offer is valid to anyone in receipt of this information and shall expire three years following the date of the final distribution of this product version by Hewlett Packard Enterprise Company. To obtain such source code, please check if the code is available in the HPE Software Center at <https://myenterpriselicense.hpe.com/cwp-ui/software> but, if not, send a written request for specific software version and product for which you want the open source code. Along with the request, please send a check or money order in the amount of US \$10.00 to:

Hewlett Packard Enterprise Company
Attn: General Counsel
WW Corporate Headquarters
1701 E Mossy Oaks Rd Spring, TX 77389
United States of America.



Acknowledgments

Intel®, Itanium®, Optane™, Pentium®, Xeon®, Intel Inside®, and the Intel Inside logo are trademarks of Intel Corporation in the U.S. and other countries.

Microsoft® and Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

Java® and Oracle® are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

All third-party marks are property of their respective owners.

Notices

The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Links to third-party websites take you outside the Hewlett Packard Enterprise website. Hewlett Packard Enterprise has no control over and is not responsible for information outside the Hewlett Packard Enterprise website.

About this document	7
Applicable products	7
Latest version available online	7
Command syntax notation conventions	7
About the examples	8
Identifying switch ports and interfaces	8
Identifying modular switch components	9
QoS overview	10
End-to-end QoS behavior	10
Best effort service	10
Class of Service	10
Differentiated services	11
QoS on the switch	12
QoS trust	14
Dynamic QoS trust mode	15
Port rate limiting	15
Queue profiles	16
Schedule profiles	17
Egress queue shaping	18
Egress port shaping	18
Active Queue Management	18
Explicit Congestion Notification (ECN)	18
Weighted Random Early Detection (WRED)	18
Threshold profiles	19
Virtual output queues	19
8400 switch series	19
Terms	19
QoS configuration	21
Configuring QoS	21
Configuring expedited forwarding for VoIP traffic	23
Configuring rate limiting	24
Configuring egress queue shaping	25
Configuring egress port shaping	26
Configuring threshold profiles	27
Monitoring queue operation	28
Data center bridging for storage and lossless Ethernet	30
Data center bridging components	30
PFC - Priority-based flow control	30
ETS - Enhanced transmission selection	30
QCN - Quantized congestion notification	31
DCBx - Data center bridging exchange protocol	31
DCBx guidelines	31
IP ECN	32
Host network interface cards	32
DCB layer 3 configuration task list	33
DCBx configuration	33

DCBx configuration considerations	34
Enabling DCBx	34
Priority-based flow control	35
QoS queue profile	36
QoS queue profile configuration	38
QoS schedule profile - enhanced transmission selection	38
DWRR Calculation	39
QoS schedule profile configuration	39
Overriding the global schedule profile on an interface	40
IP explicit congestion notification	43
IP ECN configuration considerations	44
IP ECN configuration procedure	45
Lossless QoS pool	45
Considerations and prerequisites	46
QoS pool configuration	46
QoS trust	47
Layer 3 CoS-DSCP markings	48
Verifying trust settings	49
Troubleshooting data center bridging	49

QoS commands 53

apply qos	53
apply qos threshold-profile	55
dcbx application	56
lldp dcbx (global)	57
lldp dcbx (per interface)	58
lldp dcbx disable	59
map queue	60
name queue	62
qos cos-map	62
qos dscp-map	64
qos queue-profile	65
qos schedule-profile	66
qos shape	68
qos threshold-profile	69
qos trust	70
queue action	71
rate-limit	73
show dcbx interface	75
show interface queues	79
show interface qos	80
show qos cos-map	81
show qos dscp-map	83
show qos queue-profile	84
show qos schedule-profile	85
show qos threshold-profile	86
show qos trust	88
strict queue	89
wfq queue	91

Support and Other Resources 93

Accessing HPE Aruba Networking Support	93
Accessing Updates	94
Warranty Information	94
Regulatory Information	94
Documentation Feedback	94

Chapter 1

About this document

This document describes features of the AOS-CX network operating system. It is intended for administrators responsible for installing, configuring, and managing HPE Aruba Networking switches on a network.

Applicable products

This document applies to the following products:

- HPE Aruba Networking 8400 Switch Series (JL366A, JL363A, JL687A)

Latest version available online

Updates to this document can occur after initial publication. For the latest versions of product documentation, see the links provided in [Support and Other Resources](#).

Command syntax notation conventions

Convention	Usage
<code>example-text</code>	Identifies commands and their options and operands, code examples, filenames, pathnames, and output displayed in a command window. Items that appear like the example text in the previous column are to be entered exactly as shown and are required unless enclosed in brackets ([]).
example-text	In code and screen examples, indicates text entered by a user.
Any of the following: <ul style="list-style-type: none">▪ <code><example-text></code>▪ <code><example-text></code>▪ <i>example-text</i>▪ <i>example-text</i>	Identifies a placeholder—such as a parameter or a variable—that you must substitute with an actual value in a command or in code: <ul style="list-style-type: none">▪ For output formats where italic text cannot be displayed, variables are enclosed in angle brackets (< >). Substitute the text—including the enclosing angle brackets—with an actual value.▪ For output formats where italic text can be displayed, variables might or might not be enclosed in angle brackets. Substitute the text including the enclosing angle brackets, if any, with an actual value.
	Vertical bar. A logical OR that separates multiple items from which you can choose only one. Any spaces that are on either side of the vertical bar are included for readability and are not a required part of the command syntax.
{ }	Braces. Indicates that at least one of the enclosed items is required.

Convention	Usage
[]	Brackets. Indicates that the enclosed item or items are optional.
... or ...	Ellipsis: <ul style="list-style-type: none">▪ In code and screen examples, a vertical or horizontal ellipsis indicates an omission of information.▪ In syntax using brackets and braces, an ellipsis indicates items that can be repeated. When an item followed by ellipses is enclosed in brackets, zero or more items can be specified.

About the examples

Examples in this document are representative and might not match your particular switch or environment.

The slot and port numbers in this document are for illustration only and might be unavailable on your switch.

Understanding the CLI prompts

When illustrating the prompts in the command line interface (CLI), this document uses the generic term **switch**, instead of the host name of the switch. For example:

```
switch>
```

The CLI prompt indicates the current command context. For example:

```
switch>
```

Indicates the operator command context.

```
switch#
```

Indicates the manager command context.

```
switch (CONTEXT-NAME) #
```

Indicates the configuration context for a feature. For example:

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

Identifies the **interface** context.

Variable information in CLI prompts

In certain configuration contexts, the prompt may include variable information. For example, when in the VLAN configuration context, a VLAN number appears in the prompt:

```
switch (config-vlan-100) #
```

When referring to this context, this document uses the syntax:

```
switch (config-vlan-<VLAN-ID>) #
```

Where <VLAN-ID> is a variable representing the VLAN number.

Identifying switch ports and interfaces

Physical ports on the switch and their corresponding logical software interfaces are identified using the format: **member/slot/port**.

On the HPE Aruba Networking 8400 Switch Series

- *member*: Always 1. VSF is not supported on this switch.
- *slot*: Specifies physical location of a module in the switch chassis.
 - Management modules are on the front of the switch in slots 1/5 and 1/6.
 - Line modules are on the front of the switch in slots 1/1 through 1/4, and 1/7 through 1/10.
- *port*: Physical number of a port on a line module

For example, the logical interface **1/1/4** in software is associated with physical port 4 in slot 1 on member 1.

Identifying modular switch components

- Power supplies are on the front of the switch behind the bezel above the management modules. Power supplies are labeled in software in the format: *member/power supply*:
 - *member*: 1.
 - *power supply*: 1 to 4.
- Fans are on the rear of the switch and are labeled in software as: *member/tray/fan*:
 - *member*: 1.
 - *tray*: 1 to 4.
 - *fan*: 1 to 4.
- Fabric modules are not labeled on the switch but are labeled in software in the format: *member/module*:
 - *member*: 1.
 - *member*: 1 or 2.
- The display module on the rear of the switch is not labeled with a member or slot number.

Quality of Service (QoS) enables network administrators to customize how different types of traffic are serviced on a network, taking into account the unique characteristics of each traffic type and its importance within an organization's infrastructure. QoS ensures uniform and efficient traffic handling, keeping the most important traffic moving at an acceptable speed, regardless of current bandwidth usage. It also provides methods for administrators to control the priority settings of inbound traffic arriving at each network device.

End-to-end QoS behavior

The QoS settings on each network device must be aligned to achieve the desired end-to-end QoS behavior for a network. Three service types can be used to categorize and prioritize network traffic:

- Best Effort Service
- Ethernet Class of Service (CoS)
- Internet Differentiated Services (DiffServ)

For a network as a whole, it is best to select one service type to use as the primary end-to-end behavior, and then use the other two service types as needed.

Best effort service

This is the simplest service type. All traffic is treated equally in a first-come, first-served manner. If the traffic load is low in relation to the capacity of the network links, then there is no need for the administrative complexity and costs of maintaining a more complex end-to-end policy. This is sometimes called over-provisioning, as all link speeds are much higher than peak loads on the network.

Class of Service

Class of Service (CoS) is a method for classifying network traffic at layer 2 by marking 802.1Q VLAN Ethernet frames with one of eight service classes.

CoS	Traffic type	Example protocols
7	Network Control	STP, PVST
6	Internetwork Control	BGP, OSPF, PIM
5	Voice (<10ms latency)	VoIP(UDP)
4	Video (<100ms latency)	RTP
3	Critical Applications	SQL RPC, SNMP

CoS	Traffic type	Example protocols
2	Excellent Effort	NFS, SMB
0	Best Effort	HTTP, TELNET
1	Background	SMTP, IMAP

CoS 1 is deliberately set as the lowest CoS. This enables a traffic service level below the default (best effort) traffic level to be specified.

The 3-bit Priority Code Point (PCP) field within the 16-bit Ethernet VLAN tag is used to mark the CoS.



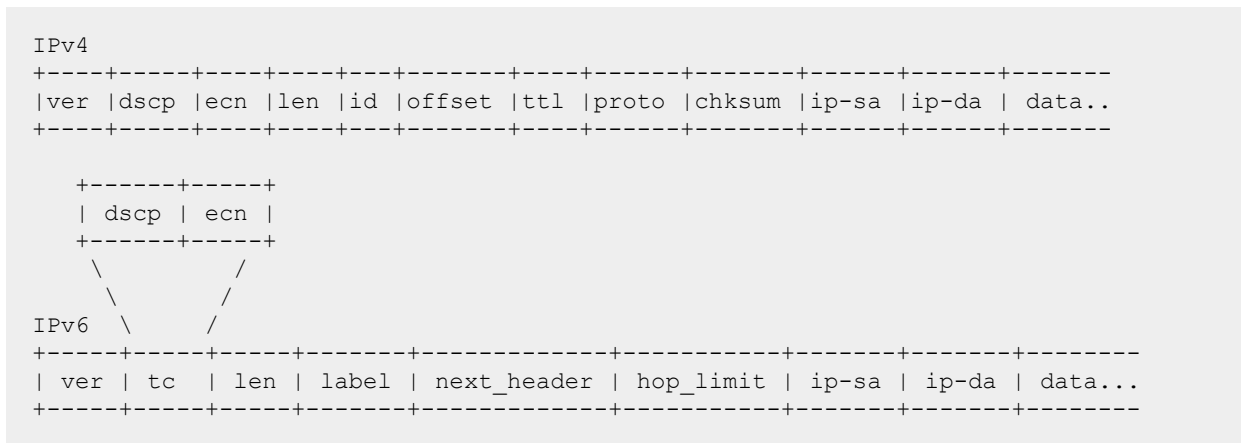
Differentiated services

Differentiated services (DiffServ) is a method for classifying network traffic at layer 3 by marking packets with one of 64 different service classes. Services classes are identified by the Differentiated services Code Point (DSCP) value. Some common DSCP values are:

DSCP	Name	Service class	RFC
48	CS6	Network Control	2474
46	EF	Telephony	3246
40	CS5	Signaling	2474
34, 36, 38	AF41, AF42, AF43	Multimedia Conferencing	2597
32	CS4	Real-Time Interactive	2474
26, 28, 30	AF31, AF32, AF33	Multimedia Streaming	2597
24	CS3	Broadcast Video	2474
18, 20, 22	AF21, AF22, AF23	Low-Latency Data	2597
16	CS2	OAM	2474
00	CS0, BE, DF	Best Effort	2474
10, 12, 14	AF11, AF12, AF13	Bulk Data	2597
08	CS1	Low-Priority Data	3662

DSCP CS1 (08) CoS 1 is deliberately set as the lowest priority. This enables a traffic service level below the standard (best effort or default forwarding) level to be specified.

The DSCP value is carried within the IPv4 DSCP field or the upper 6-bits of the 8-bit IPv6 Traffic Class (TC) field.



QoS on the switch

There are five key stages a packet passes through when traversing a switch: ingress, prioritization, destination determination, egress queuing, and transmission. The following table provides an overview of each stage, and lists the commands that can be used to configure QoS settings.

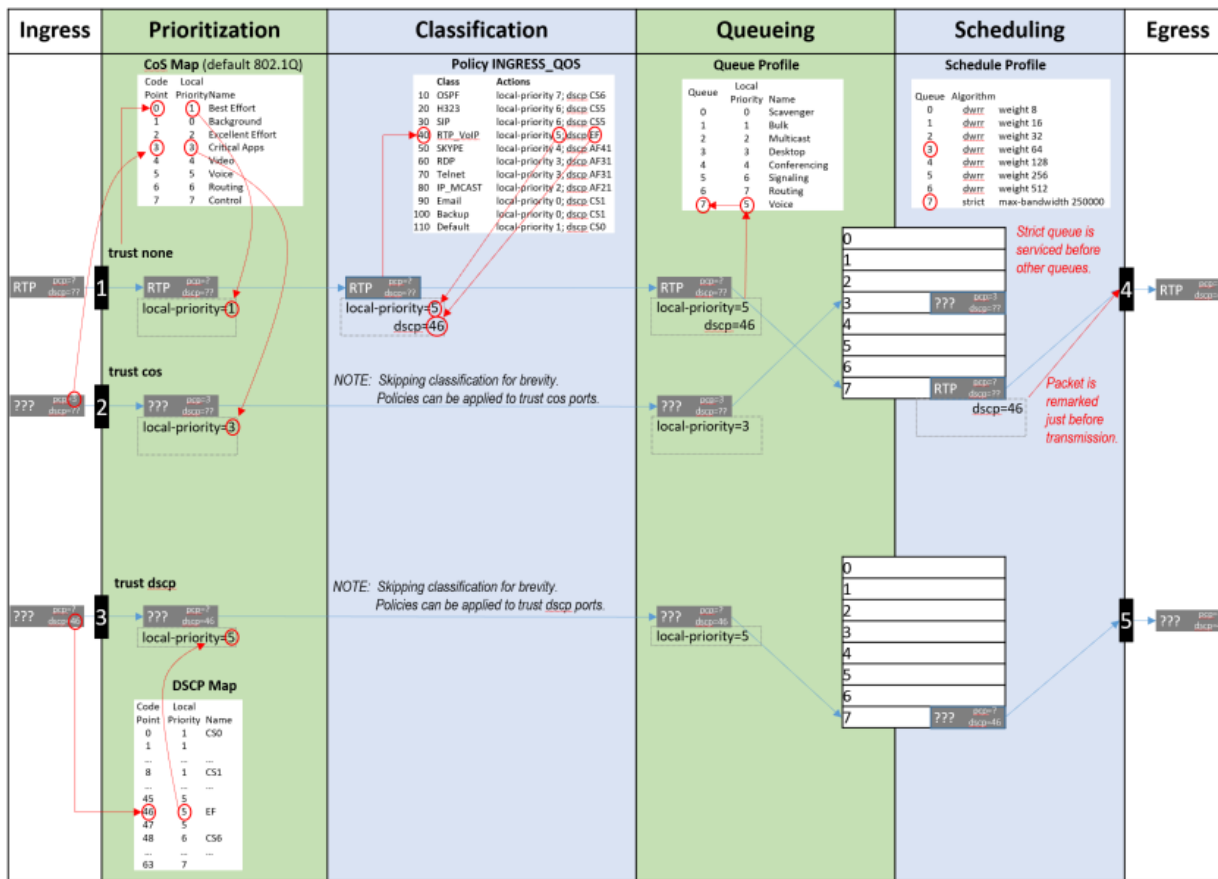


Switches with at least 52 ports will experience negative performance if a flood occurs where at least 42 ports are members of the same VLAN and all 52 ports have QoS rules applied to them.

Ingress	Prioritization	Classification	Queueing	Transmission
Packets arrive at switch interface (port).	An initial local-priority value is assigned to the packet based on VLAN CoS or IP DSCP.	Packets local-priority and code points can be remarked, and rates policed, based on many packet header fields. (see <i>ACL & Policy Guide</i>)	Packets are queued based on the destination interface and local-priority of the packet.	A scheduler defines the order in which packets are selected from queues to be transmitted from the interface.
rate-limit Rate limiting can control ingress flow by packet type: broadcast, multicast, or unknown-unicast.	qos trust {cos dscp none} Assigns which packet values are used to determine initial local-priority either globally for all interfaces, or to specific interface(s). qos cos-map Defines how CoS values are mapped to local-priority values. qos dscp-map Defines how IP DSCP values are mapped to local-priority values. qos dscp Remarks egress DSCP value.	class {ip ipv6 mac} NAME match TERMS [count] ignore TERMS [count] Creates an ordered list of rules to identify packets to match the class. policy NAME class NAME ACTIONS Creates an ordered list of classes to which traffic is evaluated, and the actions to be taken on matching packets: remarking, policing etc. apply policy NAME Assigns the policy to evaluate all traffic inbound or outbound globally, or on specific interfaces or VLANs.	qos queue-profile Creates a profile that defines transmit queues. map queue Assigns a local-priority to a queue. Packets with a matching local-priority are placed in the queue. name queue Assigns a name to a queue. qos threshold-profile Creates a profile that defines ECN limits for queues. queue NUM action ecn Assigns an ECN limit for a queue. apply qos Assigns queue, schedule, or threshold profiles globally to all interfaces, or schedule or threshold profiles to a specific interface(s).	qos schedule-profile Creates a profile that defines the order in which packets are selected from the queues for transmission. strict queue Assigns the strict priority algorithm, with optional max-bandwidth, to a queue. wfq queue Assigns the weighted WFQ algorithm to a queue. qos shape Assigns the maximum rate for traffic to be transmitted by the interface.

Ingress	Prioritization	Classification	Queueing	Transmission
Packets arrive at switch interface (port).	An initial local-priority value is assigned to the packet based on VLAN CoS or IP DSCP.	Packets local-priority and code points can be remarked, and rates policed, based on many packet header fields. (see <i>ACL & Policy Guide</i>)	Packets are queued based on the destination interface and local-priority of the packet.	A scheduler defines the order in which packets are selected from queues to be transmitted from the interface.
	show int IFNAME qos Displays all the QoS settings configured on an interface: <ul style="list-style-type: none"> Profiles Rate-limits DSCP remarks Egress shape Trust show qos cos-map Displays the contents of the CoS Map table. show qos dscp-map Displays the content of the DSCP Map table. show qos trust Display the current default trust mode.	show class {ip ipv6 mac} Displays the ordered list of the match and ignore statements in the class. show policy Displays the order list of classes in the policy. show policy hitcounts Displays the ordered list of classes in the policy. For each class the number of times that each match or ignore statement in the class matched a packet.	show int IFNAME queues Displays the per-queue counts of packets transmitted, packets dropped, bytes transmitted, and maximum byte depth. show qos queue-profile Displays the definition of queue profiles. show qos threshold-profile Displays the definition of threshold profiles.	show qos schedule-profile Displays the definition of schedule profiles.

The following diagram shows how different packets might traverse a switch. It also shows how QoS configuration settings apply at each stage.



QoS trust

Traffic priorities for networks can be carried in VLAN tags, using the CoS Priority Code Point (PCP), or in IP packet headers, using the Differentiated Services Code Point (DSCP). Whether these priorities affect how traffic is serviced, depends on how QoS trust mode is configured on the switch. QoS trust mode specifies how the switch assigns local priority values to ingress packets. Trust mode can be set globally for all interfaces, or individually for each interface. By default, trust mode is set to `none`, meaning that any QoS information in the packet (CoS or DSCP) is ignored, and local priority values are assigned from the CoS map value for code point 0.

When trust mode is set to CoS or DSCP, the switch translates the QoS settings in VLAN tags (for CoS), or the DS field in an IP header (for DSCP), to local priority values on the switch. Translation is controlled by the CoS map or DSCP map tables.

For example:

CoS Map

When QoS trust mode = CoS

	Code Point	Local Priority	Name	
Untagged packet	0	1	Best Effort	Packet local-priority = 1
	1	0	Background	
	2	2	Excellent Effort	
Tagged PCP=3 packet	3	3	Critical Apps	Packet local-priority = 3
	4	4	Video	
	5	5	Voice	
	6	6	Routing	
	7	7	Control	

DSCP Map

When QoS trust mode = DSCP

	Code Point	Local Priority	Name	
	0	1	CS0	
	1	1		
	
	8	0	CS1	
	
	45	5		
IP DSCP=46 packet	46	5	EF	Packet local-priority = 5 & DSCP = 46
	47	5		
	48	6	CS6	
	
	63	7		

When QoS trust mode = None



Dynamic QoS trust mode

The device profile feature can dynamically set the QoS trust mode on an interface based on the LLDP information exchanged with a link partner. The device profile's trust mode temporarily overrides the static trust mode configured for an interface. The override remains in place as long as that link partner is connected and its link state is **up**. Use command `show interface IFNAME qos` to view the current QoS trust mode for an interface.

Port rate limiting

Port rate limiting helps control undesirable traffic. Its purpose is to allow enough unicast, broadcast, multicast, and ICMP traffic for the network to function properly, while preventing flooding and traffic storms.

Some amount of each traffic type is required for normal network operation. For example, broadcast packets may include ARP and DHCP traffic. Video streams and certain types of network protocol packets are multicast traffic. Unknown-unicast packets may be intended for devices whose addresses have temporarily aged out of network forwarding caches. Configuring rate limits can help provide balance between necessary and flooded traffic.

The amount of permitted traffic (the rate limit) can be specified in kilobits-per second (kbps), packets-per-second (pps), or as a percentage of link bandwidth (percent).

Please note that rate limits are enforced separately on each individual member of a LAG, not on the LAG as a whole. When limits are configured in kilobits-per-second (kbps) or packets-per-second (pps), larger limits may be needed on high speed ports to allow normal network functionality. A limit of 100 packets-per-second might be reasonable on a 1 Gigabit port, but it would be too restrictive on a port that is 10 Gigabits or faster. Conversely, limits configured as a percentage of link bandwidth may not be useful on high speed ports. A limit of 1 percent might be reasonable on a 1 Gigabit port, but too lax on a port that is 10 Gigabits or faster.

A rate limit for multicast traffic will affect both multicast and broadcast traffic whereas a rate limit for broadcast traffic only impacts broadcast traffic. In the case where both types are applied to the same port, broadcast packets will be limited to the lower of the two configured rate values.

Rate limits configured on LAG interfaces with no assigned or linked-up member ports may be displayed as "not applied". This behavior may also be seen with percentage-based rate limits on regular ports when link up has not yet occurred (i.e., when the port speed is not known).

Queue profiles

A queue profile defines the queues that are associated with an interface to control the transmission of packets. Each profile supports up to eight queues, numbered 0 to 7. The larger the queue number, the higher its priority during transmission scheduling. Packets are assigned to a queue based on their local priority value (0 to 7). A queue profile must map all eight local priority values to whatever queues are being used on the switch, and a schedule profile must specify the configuration for those same queues. A queue without a local priority value assigned to it is not used to store packets.

The switch is automatically provisioned with an initial queue profile named **factory-default** which assigns each local priority to the queue of the same number. To see the default queue profile, use the command **show qos queue-profile factory-default**:

```
switch# show qos queue-profile factory-default
queue_num local_priorities name
-----
0          0                  Scavenger_and_backup_data
1          1
2          2
3          3
4          4
5          5
6          6
7          7
```

More than one local priority value can be assigned to the same queue. For example,

Local Priority	Queue
0	0
1	1
2	2
3	3
4	4

Local Priority	Queue
5	5
6	5
7	5

Queues must be consecutively defined starting at queue number zero. For example, a four-queue profile with priority values defined for queues 0, 1, 2, 3 is valid, but a four-queue profile which defines priority values for queues 1, 3, 5, and 7 is not.

Table 1: *Valid queue*

Local Priority	Queue
0	0
1	1
2	1
3	1
4	2
5	2
6	3
7	3

The commonly used commands for working with QoS queues are as follows:

- **qos queue-profile:** Creates an empty queue-profile and enters the profile configuration context.
- **name queue:** Assigns a descriptive name to a queue.
- **map queue:** Assigns a local-priority to a queue.
- **apply qos queue-profile:** Applies a queue-profile globally to all interfaces.

Schedule profiles

A schedule profile determines the order in which queues are selected for transmission, and the amount of service available for each queue. A schedule profile must be configured on every interface at all times. A schedule profile can be applied globally to all interfaces, or only to specific interfaces.

The following options are available:

- All queues use weighted fair queuing (WFQ)
- All queues use strict priority
- The highest priority queue uses strict priority, and all other queues use WFQ

A weighted schedule profile assigns relative servicing for each queue. The amount of service per weight is relative to the underlying hardware implementation, and to the weights assigned to the other non-

empty queues. Strict scheduling can be used to service queues purely on the basis of highest priority first (at the risk of starving lower-priority queues during high stress periods). A combination of strict and weighted scheduling offers more service to the highest priority queue when needed, while preserving scheduling between the remaining queues, thus decreasing the risk of starvation.

The switch is automatically provisioned with a schedule profile named **factory-default**, which assigns DWRR to all queues with a weight of 1. Use the command `show schedule-profile factory-default` to view the default schedule profile. (Do not use `show running-configuration`, as it only displays changes from the initial settings.)

```
switch# show qos schedule-profile default
queue_num algorithm weight
-----
0          wfq         1
1          wfq         1
2          wfq         1
3          wfq         1
4          wfq         1
5          wfq         1
6          wfq         1
7          wfq         1
```

Egress queue shaping

Egress queue shaping limits the amount of traffic transmitted per strict output and guaranteed minimum bandwidth queues. The buffer associated with each egress queue stores excess traffic to absorb bursts and smooths the output rate. For example, an administrator might limit strict-priority or min-bandwidth queue traffic to prevent low-priority queue starvation in the event that a device inappropriately sends too many higher-priority packets.

Egress queue shaping can be configured on an Ethernet port or on a link aggregation group (LAG). To configure egress queue shaping, define a schedule profile with the strict priority algorithm assigned to each queue.

Egress port shaping

Egress port shaping limits the amount of aggregate traffic transmitted through a port. To be effective, the egress port-shaping rate must be less than the port's line rate. By default, the egress port-shaping rate is the same as the line-rate of the port. Buffers associated with each port store excess traffic. When both egress port-shaping and egress queue-shaping are configured on the same interface, the switch respects the minimum of both configurations.

Active Queue Management

Explicit Congestion Notification (ECN)

Explicit Congestion Notification (ECN) provides a mechanism for two end-points to exchange end-to-end notification of network congestion. ECN uses a 2-bit field in the IP header to indicate that the traffic load on network equipment in the path between an ECN-capable sender and receiver is causing packets to be buffered, as defined by IETF RFC 3168 (<https://tools.ietf.org/html/rfc3168>).

Weighted Random Early Detection (WRED)

WRED operates by random early-dropping packets, which can be helpful in signaling data path congestion to certain protocols. Protocols that respond to these drops slow their transmit rate in an effort to reduce network congestion. WRED drops are randomized in order to avoid potential

synchronization between multiple streams using the same link. If drops occurred on all streams at the same time, multiple senders might respond by reducing their transmit rates and then increasing. Such synchronized behavior causes link utilization to fluctuate between high and low, wasting bandwidth. Random dropping ensures that only some streams detect drops, and that they detect them at different times. This results in better link utilization, as some senders continue to transmit at a higher rate while others reduce and ramp up again.

Threshold profiles

Threshold profiles configure individual queue utilization thresholds as triggers for taking action (i.e., ECN marking or WRED dropping) on a packet. A threshold profile is applied per-port and defines the thresholds and actions for each queue. Omitting configuration for a queue in a threshold profile means that queue will not be configured with a threshold value or action.

In an environment where responsive transport protocols are in use and congestion management features are required to reduce latency, ECN can be configured on queues carrying delay-sensitive traffic. The result is that queue utilization is actively managed, resulting in ECT packets being CE marked when queue utilization reaches or exceeds a configured threshold.

Virtual output queues

8400 switch series

The switch uses a virtual output queue (VOQ) architecture where most packet buffering occurs on the ingress line module. Traffic destined for one port (unicast), uses different buffering and scheduling than traffic destined for multiple ports (flood). The relative priority and the amount of packet transmission for these two types of traffic differ.

For unicast traffic, each line module contains eight VOQs for every destination port in the chassis (one per local priority). The queue profile determines which VOQ is used to buffer a local priority. The schedule profile determines the order of VOQ servicing. Unicast packets wait in VOQs until the scheduler selects them to cross the fabric. Each destination port has a shallow egress transmit queue that buffers unicast packets.

Broadcast, multicast, and unknown-unicast packets, collectively called flooded traffic, use a separate path to the destination ports. Each line module contains eight VOQs per destination line module (including itself) to buffer traffic to be flooded (one VOQ per local priority). A copy of the packet to be flooded is buffered in VOQs for each destination line module. The queue profile determines which VOQ is used for each local priority.

Flooded traffic VOQs use a fixed strict schedule profile to determine the order of VOQ servicing. Flooded packets wait in VOQs until the scheduler selects them to cross the fabric. On the destination line module, they are replicated to one or more destination ports. Each destination port has a second shallow egress queue for replicated packets buffered for transmit.

A WFQ scheduler is used to select packets for transmission from the unicast and replicated traffic egress queues. When selecting packets between two non-empty queues, WFQ uses a fixed data weight of four for unicast traffic, and a weight of one for replicated traffic. As long as both queues are non-empty, replicated (flooded) packets comprise approximately 20% of the transmitted traffic, independent of the unicast scheduled percentages.

Terms

Class

For networking, a set of packets sharing a common characteristic. For example, all IPv4 packets.

Code point

The name of a packet header field, or the value carried within a packet header field:

- Example 1: Priority code point (PCP) is the name of a field in the IEEE 802.1Q VLAN tag.
- Example 2: Differentiated services code point (DSCP) is the name of a field carried within the DS field of an IP packet header.

Color

A metadata label associated with each packet within the switch. It has three values: green (0), yellow (1), or red (2). When packets encounter congestion for a resource (queue), the switch uses packet color to distinguish which packets must be dropped, and is mostly used for packets marked with Assured Forwarding (AF) DSCP values.

Not supported in this release.

Class of service (CoS)

A 3-bit value used to mark packets with one of eight classes (levels of priority). It is carried within the priority code point (PCP) field of the IEEE 802.1Q VLAN tag.

Differentiated services code point (DSCP)

A 6-bit value used to mark packets for different per-hop behavior as originally defined by IETF RFC 2474. It is carried within the differentiated services (DS) field of the IPv4 or IPv6 header.

Local priority

A meta-data label associated with a packet within the switch which is used to classify packets for different treatment (such as queue assignment). Eight local priorities are defined on the switch, numbered from 0 to 7. A queue profile must map all eight local priorities to whatever queues are in use on the switch, and a schedule profile must specify the configuration for these same queues.

Metadata

Information labels associated with each packet in the switch, separate from the packet headers and data. These labels are used by the switch in its handling of the packet. For example: arrival port, egress port, VLAN membership, and local priority.

Priority code point (PCP)

The name of a 3-bit field in the IEEE 802.1Q VLAN tag. It carries the CoS value to mark a packet with one of eight classes (priority levels).

Quality of service (QoS)

General term used when describing or measuring performance. For networking, it means how different classes of packets are treated when traversing a network or device.

Traffic class (TC)

General term for a set of packets sharing a common characteristic. It used to be the name of an 8-bit field in the IPv6 header originally defined by IETF RFC 2460. This field name was changed to differentiated services by IETF RFC 2474.

Type of service (ToS)

General term when there are different levels of treatment (fare class). It used to be the name of an 8-bit field in the IPv4 header originally defined by IETF RFC 791. This field name was changed to differentiated services by IETF RFC 2474.

Configuring QoS

Procedure

1. Configure how local priority values are assigned to ingress packets with the commands `qos cos-map`, `qos dscp-map`, and `qos trust`.
2. Optionally, add a rate limit for ingress traffic on one or more interfaces with the command `rate-limit`.
3. If you do not want to use the default QoS queue profile to map local priority to queue, create one or more custom queue profiles with the command `qos queue-profile`. For each queue in a custom queue profile:
 - a. Assign a local priority value with the command `map queue`.
 - b. Optionally, define a descriptive name with the command `name queue`. All local priorities (0 to 7) must be mapped to a queue, and the queues selected for use must be in contiguous order starting at 0.
4. If you do not want to use the default QoS schedule profile to determine the order in which queues are selected to transmit a packet, create one or more custom schedule profiles with the command `qos schedule-profile`. For each queue in a custom schedule queue profile, define scheduling priority with the commands `strict queue` and `wfq queue`.
5. Optionally, create a threshold profile to limit throughput on one or more queues with the command `qos threshold-profile`. Assign threshold values to the queue with the command `queue`. Then, apply the profile with the command `apply qos threshold-profile`.
6. Optionally for strict queues, configure egress queue shaping to limit egress bandwidth on an interface to a value that is less than its line rate. Use the `max-bandwidth` parameter of the `strict queue` command.
7. Activate QoS settings with the command `apply qos`. This command lets you apply a queue profile and schedule profile globally to all interfaces, or a schedule profile override to individual interfaces.

When applying QoS settings to a port configured to support priority-based flow control, specific configuration settings must be respected when defining a CoS map and queue profile. See the command `flow-control` in the *Command-Line Interface Guide* for details.

8. View QoS configuration settings with the provided `show` commands.

Examples

This example creates the following configuration:

- Configures CoS to be used to assign local priority to ingress packets.
- Modifies the default CoS map to assign CoS 1 to local priority 1.
- Creates a queue profile named `Q1` and assigns local priorities as follows:

Queue	Local Priority
0	0
1	1
1	2
3	3
4	4
5	5
5	6
5	7

- Creates a schedule profile named `s1` and assigns WFQ to all queues in the schedule profile with the following weights:

Queue	Weight
0	5
1	10
3	15
4	25
5	50

- Creates a threshold profile named `t1` with the following limits:

Queue	Threshold
4	40 KB
5	50 KB

- Applies **Q1** and **S1** to all interfaces that do not have a QoS override applied.

```
switch(config)# qos trust cos
switch(config)# qos cos-map 1 local-priority 1
switch(config)# qos queue-profile Q1
switch(config)# map queue 0 local-priority 0
switch(config)# map queue 1 local-priority 1
switch(config)# map queue 1 local-priority 2
switch(config)# map queue 3 local-priority 3
switch(config)# map queue 4 local-priority 4
switch(config)# map queue 5 local-priority 5
switch(config)# map queue 5 local-priority 6
switch(config)# map queue 5 local-priority 7
```

```

switch(config)# qos schedule-profile S1
switch(config-schedule)# wfq queue 0 weight 5
switch(config-schedule)# wfq queue 1 weight 10
switch(config-schedule)# wfq queue 3 weight 15
switch(config-schedule)# wfq queue 4 weight 25
switch(config-schedule)# wfq queue 5 weight 50
switch(config)# qos threshold-profile T1
switch(config-threshold)# queue 4 action ecn threshold 40 kbytes
switch(config-threshold)# queue 5 action ecn threshold 50 kbytes
switch(config-threshold)# exit
switch(config)# apply qos threshold-profile T!
switch(config)# wfq queue 5 weight 50
switch(config)# apply qos queue-profile Q1 schedule-profile S1

```

Configuring expedited forwarding for VoIP traffic

Voice over IP (VoIP) traffic is delay and jitter sensitive. For optimum transmission of VoIP traffic, dwell time in network devices must be kept to a minimum and all network devices in the data path must have identical per-hop behaviors. To configure a dedicated queue on the switch to handle VoIP traffic with priority service before all other queues, follow these steps.

Prerequisites

This scenario assumes that VoIP packets are uniquely identified using DiffServ code point 46, Expedited Forwarding (EF).

Procedure

1. Map DSCP EF packets exclusively to local priority 5. The default DSCP map has eight code points (40 through 47), that are mapped to local priority 5. To reserve local priority 5 for VoIP traffic, the other code points must be reassigned. In this scenario, local priority 6 is used for all reassignments, including for code point 40, Call Signaling protocol (CS5).

```

switch(config)# qos dscp-map 40 local-priority 6 name CS5
switch(config)# qos dscp-map 41 local-priority 6
switch(config)# qos dscp-map 42 local-priority 6
switch(config)# qos dscp-map 43 local-priority 6
switch(config)# qos dscp-map 44 local-priority 6
switch(config)# qos dscp-map 45 local-priority 6
switch(config)# qos dscp-map 47 local-priority 6

```

2. Queue 7 is the highest priority queue, so for best throughput, create a queue profile that maps local priority to queue 7.

```

switch(config)# qos queue-profile ef_priority
switch(config-queue)# name queue 7 Voice_Priority_Queue
switch(config-queue)# map queue 7 local-priority 5
switch(config-queue)# map queue 6 local-priority 7
switch(config-queue)# map queue 5 local-priority 6
switch(config-queue)# map queue 4 local-priority 4
switch(config-queue)# map queue 3 local-priority 3
switch(config-queue)# map queue 2 local-priority 2

```

```
switch(config-queue) # map queue 1 local-priority 1
switch(config-queue) # map queue 0 local-priority 0
```

3. Create a schedule profile that services queue 7 using strict priority (SP), and the remaining queues with WFQ. This scenario gives all WFQ queues equal weight.

```
switch(config) # qos schedule-profile voip
switch(config-schedule) # strict queue 7
switch(config-schedule) # wfq queue 6 weight 1
switch(config-schedule) # wfq queue 5 weight 1
switch(config-schedule) # wfq queue 4 weight 1
switch(config-schedule) # wfq queue 3 weight 1
switch(config-schedule) # wfq queue 2 weight 1
switch(config-schedule) # wfq queue 1 weight 1
switch(config-schedule) # wfq queue 0 weight 1
switch(config-schedule) # exit
switch(config) #
```

4. Apply the profiles to all interfaces.

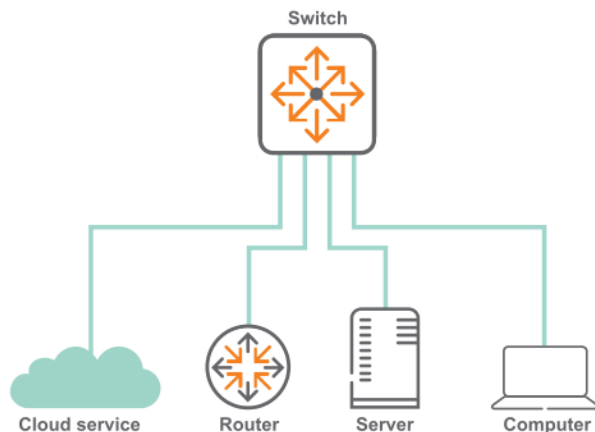
```
switch(config) # apply qos queue-profile ef_priority schedule-profile voip
```

5. Configure DSCP trust mode on all ports

```
switch(config) # qos trust dscp
```

Configuring rate limiting

This scenario illustrates how to use rate limiting to manage the traffic from various devices connected to a switch. The physical topology of the network looks like this:



A certain amount of broadcast traffic is necessary to maintain healthy network operation, particularly from routers and across service boundaries. In this scenario, both the service cloud and the router connections limit this traffic to 1 Gbps. The server has a smaller limit, as it does not require as much network protocol traffic as the service cloud and router.

A multicast server needs to be able to stream multicast traffic to clients, so a multicast rate limit may not be helpful. A computer, however, should not be generating large amounts of multicast traffic (it may

be receiving streams, but typically not sending them). In this example, the computer is configured with a multicast rate limit to prevent malicious traffic from taking up network bandwidth.

Finally, while the service cloud and router may need to send traffic for unknown unicast addresses to resolve address forwarding, the server and computer should send very little of this type of traffic. Rate limiting unknown unicast traffic on those two devices enforces that.

Procedure

1. Configure broadcast and multicast rate limiting for the service cloud connection.

```
switch# config
switch(config)# interface 1/1/1
switch(config-if)# rate-limit broadcast 1000000 kbps
switch(config-if)# rate-limit multicast 2000000 kbps
switch(config-if)# exit
```

2. Configure broadcast rate limiting for the router connection.

```
switch(config-if)# interface 1/1/2
switch(config-if)# rate-limit broadcast 1000000 kbps
switch(config-if)# exit
```

3. Configure broadcast and unknown unicast rate limiting for the server connection.

```
switch(config-if)# interface 1/1/5
switch(config-if)# rate-limit broadcast 500000 kbps
switch(config-if)# rate-limit unknown-unicast 500 kbps
switch(config-if)# exit
```

4. Configure broadcast, unknown unicast, and multicast rate limiting for the computer connection.

```
switch(config-if)# interface 1/1/10
switch(config-if)# rate-limit broadcast 1000 kbps
switch(config-if)# rate-limit multicast 500 kbps
switch(config-if)# rate-limit unknown-unicast 200 kbps
```

Configuring egress queue shaping

This example shows how to apply egress queue shaping to an interface. First, a schedule profile is created that has per-queue bandwidth limits set on all queues with `strict` as the scheduling algorithm. Next, this profile is applied to an interface or LAG.

The following example creates a schedule profile named **EQSEExample**, which services all queues using **strict** priority. This profile configures queues 1, 4, and 7 with a bandwidth limit of 10 Gbps, 20 Gbps, and 30 Gbps respectively. Queues 1 and 7 are also configured with a burst of 30 KB (burst configuration is only supported on the 8320, 8325, 9300, and 10000). The **apply qos schedule-profile** command is then used to apply the **EQSEExample** profile to interface **1/1/1**. The actual burst and bandwidth configured on an interface can be found by using the **show interface <IF-NAME> qos** command.

```

switch(config)# qos schedule-profile EQSExample
switch(config-schedule)# strict queue 0
switch(config-schedule)# strict queue 1 max-bandwidth 10000000 kbps burst 30
switch(config-schedule)# strict queue 2
switch(config-schedule)# strict queue 3
switch(config-schedule)# strict queue 4 max-bandwidth 20000000 kbps
switch(config-schedule)# strict queue 5
switch(config-schedule)# strict queue 6
switch(config-schedule)# strict queue 7 max-bandwidth 30000000 kbps burst 30
switch(config-schedule)# exit
switch(config)# interface 1/1/1
switch(config-if)# apply qos schedule-profile EQSExample

```

The following example creates a schedule profile named **EQSExample** which services all queues using **strict** priority. This profile configures queues 1, 4 and 7 with a bandwidth limit of 10%, 20%, and 30% of link bandwidth respectively. Queues 1 and 7 are also configured with a burst of 30 KB. The **apply qos schedule-profile** command is then used to apply the **EQSExample** profile to the port. The actual burst and bandwidth configured on an interface can be found by using the **show interface <IF-NAME> qos** command.

```

(config)# qos schedule-profile EQSExample
(config-schedule)# strict queue 0
(config-schedule)# strict queue 1 max-bandwidth 10 percent burst 30
(config-schedule)# strict queue 2
(config-schedule)# strict queue 3
(config-schedule)# strict queue 4 max-bandwidth 20 percent
(config-schedule)# strict queue 5
(config-schedule)# strict queue 6
(config-schedule)# strict queue 7 max-bandwidth 30 percent burst 30

```

Configuring egress port shaping

This example shows how to apply egress port shaping to an interface to limit the rate of egress traffic. Egress port shaping is configured by specifying the desired bandwidth rate in kilobits per second (kbps) or as a percentage of link bandwidth with an optional burst size in kilobytes (KB). In order to be effective, the configured rate must be less than the link rate of the port. If the configured egress port-shaping rate exceeds the port's link rate, then egress port shaping is effectively disabled. Egress port shapes are enforced separately on each individual member of a LAG, not on the LAG as a whole. Because shapes are in kilobits-per-second (kbps), larger shapes may be needed on high speed ports in order to allow normal network function. A shape of 100 kbps might be reasonable on a 1-Gigabit port, but too restrictive on a port that is 10-Gigabits or faster.

The configured egress rate and burst on a specific interface can be found by using the **show interface** and **show interface qos** commands.

The following example configures an egress rate of 100 Mbps and a burst of 30 KB:

```

switch(config)# interface 1/1/1
switch(config-if)# qos shape 100000 kbps burst 30

```

In the next example, both egress port shaping and egress queue shaping are configured on the same interface.

The example creates a schedule profile named **EQSExample** with strict priority for all seven queues. Queue 7 is configured with a bandwidth limit of 300 Mbps and a burst of 30 KB. The profile is then

applied to interface 1/1/1 with egress port shaping of 400 Mbps and a burst size of 30 KB. As egress queue shaping and egress port shaping are both configured on port 1/1/1, egress queue shaping is subject to the lower port or queue shape rate. The effective bandwidth for the traffic egressing on queue 7 will be 300 Mbps and the effective burst will be 30 KB.

```
switch(config)# qos schedule-profile EQSExample
switch(config-schedule)# strict queue 0
switch(config-schedule)# strict queue 1
switch(config-schedule)# strict queue 2
switch(config-schedule)# strict queue 3
switch(config-schedule)# strict queue 4
switch(config-schedule)# strict queue 5
switch(config-schedule)# strict queue 6
switch(config-schedule)# strict queue 7 max-bandwidth 300000 kbps burst 30
switch(config-schedule)# exit
switch(config)# interface 1/1/1
switch(config-if)# apply qos schedule-profile EQSExample
switch(config-if)# qos shape 400000 kbps burst 30
```

Configuring an egress port shape of 10 percent and a burst of 30 KB on port 1/1/1. The configured egress rate and burst on a specific interface can be found by using the **show interface** and **show interface qos** commands.

```
switch(config)# interface 1/1/1
switch(config-if)# qos shape 10 percent burst 30
```

Configuring threshold profiles

The threshold profile is an optional configuration that specifies a per-packet action to be taken for each port queue when the queue utilization reaches or exceeds the configured threshold. The per-queue configuration within the threshold profile is optional, which means any number of queues may be configured. An unconfigured queue means that no threshold action behavior will occur. Also, an empty threshold profile is valid and can be applied.

Use the `apply qos threshold-profile <THRESHOLD-NAME>` command in the global context to configure all ports to use the profile, or in the interface context to configure the interface to use the profile. No threshold profiles are created or configured by default.

Each entry in a threshold profile can configure an ECN action or a WRED action, either of which can take effect depending on the contents of the packet being added to the queue. For WRED action, it is possible to configure separate thresholds.

In an environment where congestion management features are required in order to reduce latency and responsive transport protocols are in use, ECN can be configured on queues carrying delay-sensitive traffic. The desired result is that a network device's port-queue utilization is actively managed, resulting in ECN-capable transport (ECT) packets being Congestion Encountered (CE) marked when queue utilization reaches or exceeds a configured threshold.

Procedure

To configure one or more port queues with ECN or WRED, complete the following steps:

1. Create a threshold profile defining one or more queue threshold configurations with an ECN or WRED action.

Create a threshold profile with an ECN or WRED action on queue 7:

```
switch(config)# qos threshold-profile threshprofile
```

ECN:

```
switch(config-threshold)# queue 7 action ecn all threshold 40 kbytes
```

WRED:

```
switch(config-threshold)# queue 7 action wred yellow min-threshold 400  
kbytes max-threshold 1000 kbytes max-prob 30 percent
```

2. Apply the threshold profile globally (all ports), OR apply the Threshold Profile as a port override to any ports that require ECN or WRED behavior.

(Option 1) Apply the threshold profile as a global default (all ports):

```
switch(config)# apply qos threshold-profile threshprofile
```

(Option 2) Apply the threshold profile to specific Ethernet or LAG interfaces as a port override:

```
switch(config)# int 1/1/1  
switch(config-if)# apply qos threshold-profile threshprofile  
switch(config)# int 1/1/2  
switch(config-if)# apply qos threshold-profile threshprofile  
switch(config)# int lag 10  
switch(config-if)# apply qos threshold-profile threshprofile
```

Monitoring queue operation

Use the show interface queues command to display the traffic transmitted per queue, and the number of packets dropped due to the queue being full. For example:

```
switch# show interface 1/1/5 queues  
Interface 1/1/5 is up  
Admin state is up  
      Tx Bytes      Tx Packets      Tx Drops      Tx Byte Depth  
Q0      157113373520      1890863919          0          1362  
Q1      233312143017      2808451320         18          65550  
Q2      156814056423      1887257650          0          1392  
Q3      157441358980      1894815504          0          1374  
Q4      157700809294      1897941370          0          1362  
Q5      157872849381      1900014146          0          1392  
Q6      183486049854      2208268429          0          4398  
Q7      231607534141      2787913734          0          65544
```

- **Tx Bytes:** Total bytes transmitted. The byte count may include packet headers and internal metadata that are removed before the packet is transmitted. Packet headers added when the packet is transmitted may not be included.
- **Tx Packets:** Total packets transmitted.
- **Tx Drops:** The number of packets dropped by a queue before it was sent. When traffic cannot be forwarded out an egress interface, it backs up at ingress. The more servicing assigned to a queue by a schedule profile, the less likely traffic destined for that queue will back up and be dropped. Tx Drops shows the sum of packets that were dropped across all line modules (due to insufficient capacity) by the ingress Virtual Output Queues (VOQs) destined for the egress port. As the counts are read separately from each line module, the sum is not an instantaneous snapshot.
- **Tx Byte Depth:** Largest byte depth (or high watermark) found on any ingress line module Virtual Output Queue (VOQ) destined for the egress port.



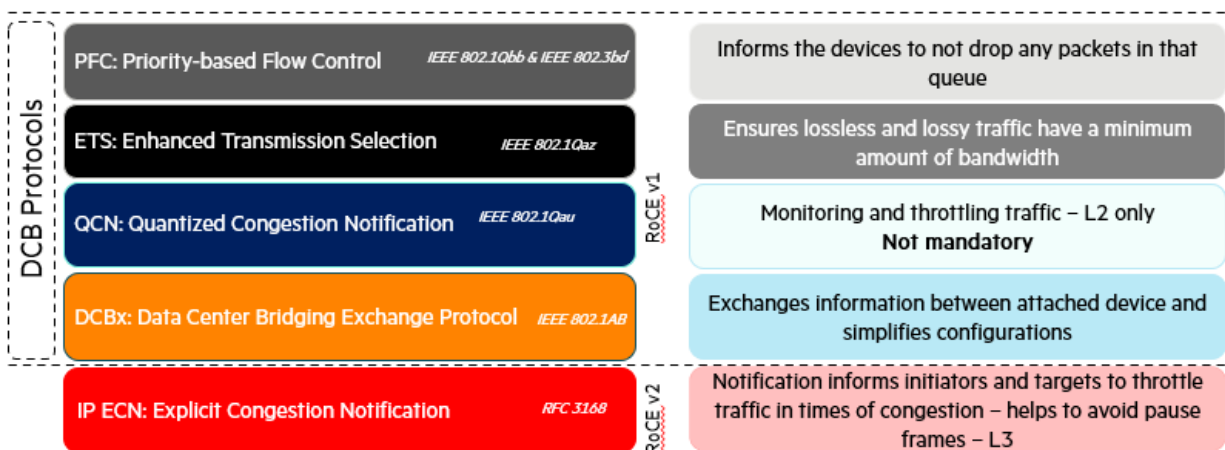
Supported on the 8400 Switch Series.

The term *lossless Ethernet* is intertwined with the Data Center Bridging protocol suite (DCB) to eliminate traffic loss within an Ethernet fabric. In a lossy fabric, traffic loss can occur due to queue buffer overflow. DCB standards define methods for avoiding packet drops, head of line blocking, and excessive latency, which can be applied to different priorities of traffic flowing on an interface. When these settings are applied uniformly across an Ethernet fabric, lossless Ethernet can be achieved.

DCB achieves this by managing bandwidth, priority, and flow control of selected traffic priorities when sharing the same Ethernet network link.

Data center bridging components

Data center bridging is comprised of the following key components:



IP ECN is an extension to IP supporting TCP and is covered by RFC 3168. It is a requirement to support lossless traffic flows over L3 networks.

PFC - Priority-based flow control

Priority-based Flow Control (PFC) is a link-level flow control mechanism similar to Ethernet PAUSE and standardized in IEEE 802.1Qbb.

The standard Ethernet PAUSE stops all traffic on an Ethernet link impacting all traffic flows. PFC PAUSE stops only the priorities specified by the data within the pause frame, allowing other traffic priorities to continue unaffected.

ETS - Enhanced transmission selection

Enhanced Transmission Selection (ETS) allows management of the link bandwidth by specifying bandwidth availability to each priority group. Priorities can be grouped according to bandwidth or bandwidth allocated to each individual priority.

QCN - Quantized congestion notification

Quantized Congestion Notification (QCN) provides congestion control at Layer 2. It was developed in parallel to the related effort on PFC.

QCN is designed to provide congestion awareness to an upstream link partner, allowing the link partner to reduce the traffic rate in order to avoid congestion on the downstream device.



This standard is not required if the PFC feature is available and is not supported on the CX switch platform.

DCBx - Data center bridging exchange protocol

Data Center Bridging Exchange protocol (DCBx) is a discovery and capability exchange protocol for communicating Data Center Bridging configuration information between link peers. DCBx is specified as part of IEEE 802.1Qaz-2011. DCBx uses LLDP as the underlying protocol for exchange of parameters with the peer.

The DCBx parameters are exchanged as LLDP TLVs. AOS-CX switches support both pre-standard CEE and IEEE standard DCBx.

DCBx supports VSX synchronization. For more information about enabling VSX synchronization, see the *Virtual Switching Extension (VSX) Guide* for your switch and software version.

DCBx is an extension of LLDP used by DCB devices to exchange configuration information with directly connected peers. Its applications include:

- Discovery of peer capabilities
- Detection of misconfigurations and identifying configuration mismatches
- Advising a DCB peer of which DCB configurations it should adopt

DCBx guidelines

- DCBx is disabled by default.
- LLDP must be enabled on the interfaces supporting DCBx.
- DCBx should be configured on all L2 interfaces.
- L3 hops which leverage PFC and ECN do not require DCBx, however it is recommended to enable it.
- DCBx is only supported on physical interfaces and not on management or logical interfaces, similar to how LLDP behaves.
- DCBx is not essential between host and switch but is highly desirable as to avoid manual PFC configurations of host devices.
- Supported protocols include:
 - DCBx Converged Enhanced Ethernet (CEE)
 - DCBx IEEE 802.1Qaz
- AOS-CX advertises DCBx with the willing bit set to 0 in all TLVs. This tells the peer that the switch is not willing to change its configuration to match the peer's configuration.
- When a peer device does not support the configured DCBx version (IEEE or CEE), a misconfiguration error will be displayed in the `show dcbx interface output`.

IP ECN

IP Explicit Congestion Notification (ECN) is not part of the DCB protocol standard and is covered separately by RFC 3168, however ECN plays a critical role in enhancing lossless Ethernet over a layer 3 IP fabric. IP ECN is a congestion notification protocol which throttles back traffic rather than pausing or dropping packets in the event of congestion. It is a slow acting mechanism and assists in avoiding pause frames generated by PFC by throttling back traffic before queue congestion occurs.

When devices in an IP ECN fabric experience congestion, the device marks congestion via ECN bits in the IP DiffServ field and forwards the IP packets on to either the next IP hop or to the end host destination. The receiving host is IP ECN aware and the congestion notification is handled by the upper layer protocols such as TCP. The IP ECN congestion bits are then echoed back to the transmitting host device which in turn reduces its transmission rate.

In order for IP ECN to work, the sender and receiver's upper layer protocol must be congestion-aware such as TCP or SCTP (Stream Control Transmission Protocol). In addition, all switch hops in the path between the sender and receiver are required to be ECN-aware and configured appropriately.

Host network interface cards

Network interface cards (NICs) on hosts that require lossless Ethernet can support the following protocols:

- PFC
- ETS
- DCBx*
- IP ECN



*Support for DCBx is not mandatory but highly desirable.

PFC support is a prerequisite for storage and lossless Ethernet networking. AOS-CX provides the following DCB features which are supported across specific listed CX platforms:

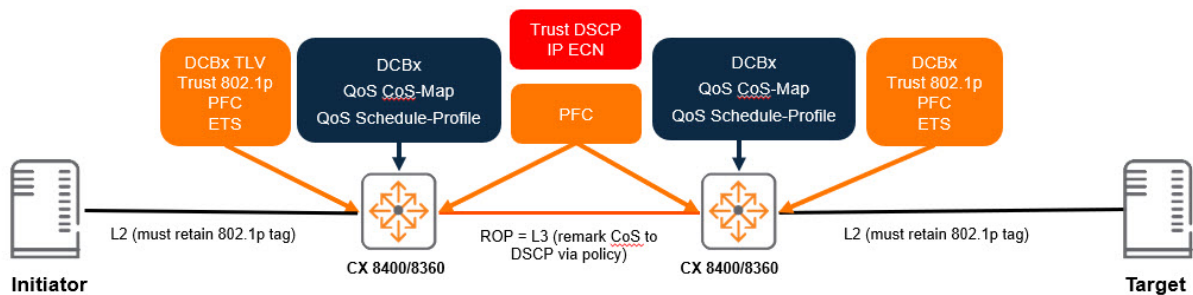
Feature	AOS-CX 10.10 or later
DCB Protocols	Priority-based Flow Control (PFC) Yes (10000/9300/8400/8100/8360/8325)
	Enhanced Transmission Selection (ETS) Yes (10000/9300/8400/8100/8360/8325)
	Data Center Bridging Exchange (DCBx) Yes (10000/9300/8400/8100/8360/8325)
	Quantized Congestion Notification (QCN) No
L3 Notification Protocol	IP Explicit Congestion Notification (ECN) Yes (10000/9300/8400/8100/8360/8325)
Lossless PFC priorities per port	7 (10000/9300/8325) 2 (8100/8360) 1 (8400)
Lossless pools	3 (10000/9300/8325)



DCB protocols are only supported on the network underlay and are not supported across VXLAN overlay.

DCB layer 3 configuration task list

1. Enable DCBx
2. Configure and apply QoS queue and schedule profiles
3. Optionally make any required configuration changes to the CoS and DSCP maps.
4. Configure global trust and override trust settings on individual interface settings
5. Enable PFC on the required priorities for each interface where it is necessary
6. Configure DCBx application TLVs
7. Configure ECN



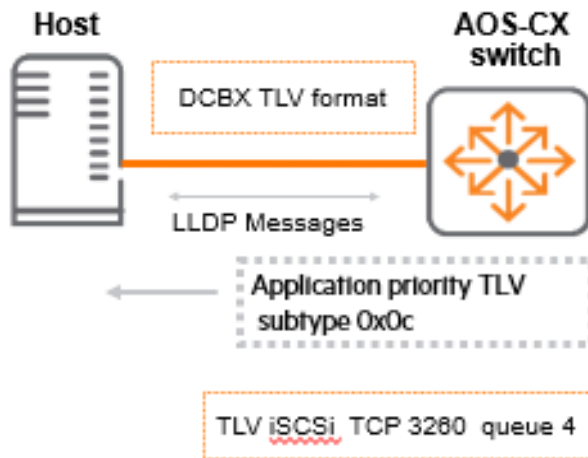
DCBx configuration

DCBx is dependent on LLDP and uses LLDP TLVs to exchange configuration details. A DCBx-enabled device can advertise various parameters including ETS, PFC, and 802.1p CoS value.

A DCBx-enabled switch can request its peer to map certain application traffic to a priority using the DCBx TLV subtype (0x0c). Whether the host accepts the QoS parameters or rejects them is dependent on the host's local DCBx NIC willing "W" bit state:

- If set to 1 - Indicates that the adapter is willing to accept QoS parameters from the remote peer
- If set to 0 - Operational QoS parameters are always resolved from the local Host nic QoS parameters

An example of DCBx TLV advertisement where the switch advertises to the host to use TCP port 3260 for iSCSI and places it in queue 4 (802.1p CoS value 4):



DCBx configuration considerations

DCBx is expected to operate over a point-to-point link. If multiple LLDP neighbors are detected, then DCBx behaves if its peer DCBx TLVs are not present and behaves this way until the multiple LLDP neighbor condition is no longer present.

- DCBx can be enabled globally or within a specific interface and is disabled by default
- The IEEE DCBx IEEE 802.1Qaz is the default version when DCBx is enabled
- LLDP must be enabled prior to configuring DCBx
- DCBx is not essential between host and switch but is highly desirable as to avoid manual PFC configurations of host devices.
- DCBx should be configured on every switch hop interface.

Enabling DCBx

LLDP must be enabled globally prior to any DCBx configuration and can be enabled with the `lldp enable` command. DCBx can be enabled either globally or at the interface level and can specify either the IEEE or CEE version by using the `lldp dcbx` command.



The DCBx application TLV is part of the DCBx configuration but must be configured after the PFC configuration.

Enabling DCBx globally:

```
switch(config)# lldp dcbx
```

Enabling the IEEE version of DCBx globally:

```
switch(config)# lldp dcbx version ieee
```

Enabling the CEE version of DCBx globally:

```
switch(config)# lldp dcbx version cee
```

Enabling DCBx for an interface:

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

Enabling the IEEE version of DCBx for an interface:

```
switch(config-if)# lldp dcbx version ieee
```

Enabling the CEE version of DCBx for an interface:

```
switch(config-if)# lldp dcbx version cee
```

The `no` form of any of these commands will remove the configuration from the switch. In order to do this you must first enter the appropriate configuration context as indicated by the `switch(config)#` prompt for global configuration or `switch(config-if)#` for interface configuration. For example, removing the IEEE version of DCBx from a specific interface:

```
switch(config)# interface 1/1/1  
switch(config-if)# no lldp dcbx version ieee
```

Priority-based flow control

Priority-based Flow Control (PFC) provides an enhancement to the Ethernet flow control pause command. PFC operates at layer 2, supporting layer 2 and layer 3 network connectivity. The standard Ethernet pause frame will pause all traffic on an Ethernet link when an ingress interface buffer is full, impacting all traffic flows.

The 802.1Qbb standard for PFC allows up to 8 different traffic classes to separately be paused by a device. AOS-CX switch devices support a varying number of PFC-enabled priorities on each port, depending on the model. However, a priority configured with PFC should be assigned to its own queue. Never combine more than one PFC priority in a queue because either priority being paused will result in both getting paused by the switch. The PFC receiving side sends a pause frame (XOFF) to the transmitting station when the buffer becomes full for a specific PFC-configured traffic class. This pause frame requests the transmitting station stop sending packets of that traffic class but allows all other traffic to continue.

It is recommended to configure PFC or link-level flow control RXTX mode and pools once on a running system. Changing the pool or flow control configuration on a running system will cause brief traffic disruption and packets will be lost in the data flow while the reconfiguration occurs. This may affect one, multiple, or all ports, depending on the existing configuration, and changes made to it.

Switch series

Priority-based flow control priorities

8325/8325H/9300/10000

Up to 7 PFC priorities per interface.

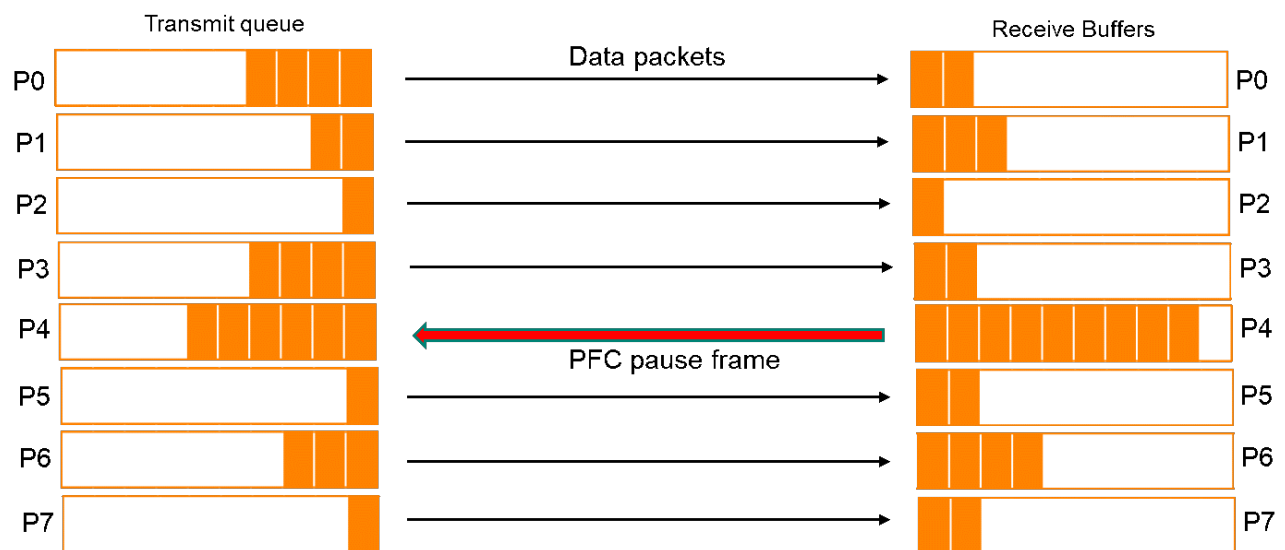
Switch series	Priority-based flow control priorities
8360	Up to 2 PFC priorities per interface.
8400	One PFC priority configured per interface.

PFC has the following characteristics:

- IEEE 802.1Qbb is based on Ethernet flow control
- Pause frames stop transmitters when downstream buffer is full
- Required to support lossless Ethernet (avoid dropping storage traffic)
- Provides mechanism with PAUSE per 802.1p priority
- Provides PAUSE of one or more traffic classes



Jumbo frame MTU size is required on network infrastructure to support storage networking traffic.



QoS queue profile

The QoS queue profile maps local priorities to queues. The CoS-map defines the 802.1p priority code mappings to the local priorities. The relationship to the QoS queue profile for lossless Ethernet is the CoS map and the DSCP map. The factory defaults are applied for both maps at system start up.

Default CoS map

```
switch# show qos cos-map default
code_point local_priority color name
-----
0          1              green Best_Effort
1          0              green Background
2          2              green Excellent_Effort
3          3              green Critical_Applications
4          4              green Video
5          5              green Voice
6          6              green Internetwork_Control
```

```
7          7          green  Network_Control
```

```
switch# show qos dscp-map
```

DSCP	code_point	local_priority	mpls_exp	color	name
000000	0	1	0	green	CS0
Codepoints 1-7 removed for brevity					
001000	8	0	1	green	CS1
Codepoints 9-15 removed for brevity					
010000	16	2	2	green	CS2
Codepoints 17-23 removed for brevity					
011000	24	3	3	green	CS3
Codepoints 25-31 removed for brevity					
100000	32	4	4	green	CS4
Codepoints 33-39 removed for brevity					
101000	40	5	5	green	CS5
Codepoints 40-47 removed for brevity					
110000	48	6	6	green	CS6
Codepoints 48-55 removed for brevity					
111000	56	7	7	green	CS7
Codepoints 57-63 removed for brevity					

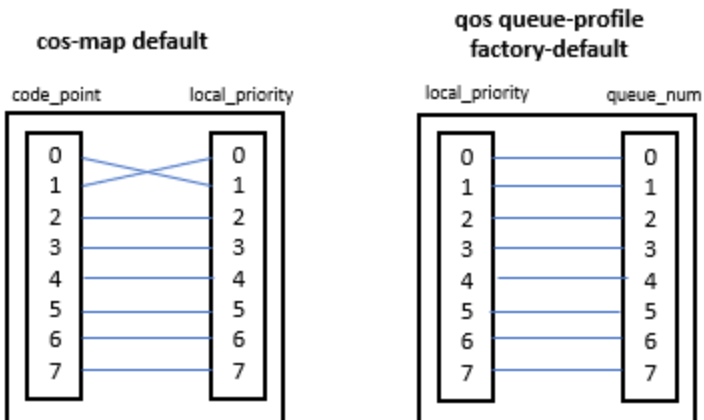
Factory default QoS queue profile

A single factory-default queue profile is initially applied to every port:

```
switch# show qos queue-profile factory-default
```

queue_num	local_priorities	name
0	0	Scavenger_and_backup_data
1	1	
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	

The default queue-profile has each local-priority mapped to a separate queue:



Best effort traffic will use Queue 1 using the default cos-map and qos-queue profile.

- Code point 0 (CoS value 0) is mapped to local priority 1 in the default cos-map for best effort traffic
- Code point 1 (CoS value 1) is mapped to local priority 0 in the default cos-map for less than best effort traffic
- Local priority 1 is mapped to queue 1 in the qos queue-profile factory-default
- The default CoS map and DSCP map can be leveraged noting the CoS value 0 (typically best effort traffic) will be mapped to queue 1 when using the default maps

A QoS queue profile is always applied to the switch and is necessary to specify internal priority to queue mapping. At all times either the factory-default queue profile or a user-specified queue profile is applied to the switch.

QoS queue profile configuration

The preferred QoS queue profile configuration will depend on the number PFC priorities as part of the lossless traffic class and the overall QoS requirements for all traffic classes.

- A local priority can only be associated with a single queue
- Any number of local priorities can be configured to a single queue
- It is essential that local priority 7 is given a dedicated queue with a minimum bandwidth for infrastructure control traffic to avoid buffer starvation from other priority queues



It is not recommended to assign multiple local priorities to the same queue if those local priorities are used for lossless traffic.

QoS queue profile configuration involves the following steps:

1. Identify local priorities for mapping to the QoS queue profile
2. Map local priorities to the appropriate queue

Create a QoS queue profile called `queue-profile-1`:

```
switch(config)# qos queue-profile queue-profile-1
```

Map queues in `queue-profile-1` to local priorities:

```
switch(config)# qos queue-profile queue-profile-1
switch(config-queue)# map queue 0 local-priority 0
switch(config-queue)# map queue 1 local-priority 1,2,5,6
switch(config-queue)# map queue 2 local-priority 3
switch(config-queue)# map queue 3 local-priority 4
switch(config-queue)# map queue 4 local-priority 7
```

The no form of the `qos queue-profile` command will remove the QoS queue profile:

```
switch(config)# no qos queue-profile queue-profile-1
```

QoS schedule profile - enhanced transmission selection

AOS-CX leverages the WFQ algorithm to provide different weights to each QoS queue which will support a single or a number of traffic classes in a profile. The profile is called the schedule-profile and by default all interfaces will have the default-schedule profile applied.



The configuration examples below apply to the 8100, 8325, 8360, 9300, and 10000 Switch Series. Configuration on the 8400 Switch Series is similar with the only difference being it uses the WFQ algorithm instead of DWRR.

The factory default schedule profile :

```
switch# show qos schedule-profile factory-default
queue_num algorithm      weight max-bandwidth_kbps burst_KB
-----
0          dwrr                1
1          dwrr                1
2          dwrr                1
3          dwrr                1
4          dwrr                1
5          dwrr                1
6          dwrr                1
7          dwrr                1
```

DWRR Calculation

To calculate the bandwidth relative to the DWRR weight, the queue weight is divisible by the sum of the weights for all queues. The default schedule profile has 8 queues with each queue having a weight of 1. To calculate the bandwidth used for each queue, the individual queue weight is divided by the sum of weights of all queues. An example using the factory default schedule profile which represents the minimum bandwidth for each queue:

- Queue 0 dwrr weight $1 - 1(8) = 12.5\%$
- Queue 1 dwrr weight $1 - 1(8) = 12.5\%$
- Queue 2 dwrr weight $1 - 1(8) = 12.5\%$
- Queue 3 dwrr weight $1 - 1(8) = 12.5\%$
- Queue 4 dwrr weight $1 - 1(8) = 12.5\%$
- Queue 5 dwrr weight $1 - 1(8) = 12.5\%$
- Queue 6 dwrr weight $1 - 1(8) = 12.5\%$
- Queue 7 dwrr weight $1 - 1(8) = 12.5\%$



The total percentage should add up to 100% for all weights.

For lossless Ethernet traffic flows, a custom schedule profile must provide the appropriate amount of bandwidth to the lossless queues as well as the lossy queues configured on the switch system. The queue-profile and the schedule-profile must specify the same queues. If the custom schedule-profile uses the same queues as the factory-default queue-profile, then no custom queue-profile is required.

QoS schedule profile configuration

The following considerations should be taken into account when configuring a QoS schedule profile:

- Assign WFQ weights based on lossless flows, traffic flow volume, and a consideration to other traffic classes.

- Ensure that the highest numbered queue specified in the applied queue-profile (assigned to critical networking traffic) has a minimum amount of bandwidth to avoid queue starvation.
- Number of queues
 - On the 9300 Switch Series the number of queues in the queue profile must align to the number of queues with allocated weights in the schedule profile.
 - On the 8325/10000 Switch Series it is possible to configure any of the 8 queues (0-7). The number of queues as well as the queue numbers themselves must be the same between the applied queue profile and schedule profile.
- It is recommended to use weights totaling 100 for easy reference to bandwidth percentage reservations.

QoS schedule profile configuration involves the following steps:

1. Configure QoS queue profile
2. Create the schedule profile
3. Apply the QoS queue profile and schedule profile

Creating a QoS schedule profile:

```
switch (config)# qos schedule-profile schedule-profile1
```

The below example shows an ETS configuration within a 2-queue environment. The settings use a weight to set the amount of available bandwidth for each queue. The settings in the example below would ensure that 50% of bandwidth will be applied to both queue 0 and queue 1:

```
switch(config)# qos schedule-profile schedule-profile1
switch(config-schedule)# dwrr queue 0 weight 15
switch(config-schedule)# dwrr queue 1 weight 15
```

Applying the schedule profile and queue profile:

```
switch (config)# apply qos queue-profile profile1 schedule-profile schedule-profile1
```

The global configuration will be applied to all interfaces.

Overriding the global schedule profile on an interface

A different schedule profile can be applied directly to interfaces which will override the global schedule profile. The caveat is that the schedule profile queues must align to the same queues as defined in the applied QoS queue profile. The procedure for applying a different schedule profile to a specific interface is as follows:

1. Create a schedule profile that specifies the same queues used in the globally-applied queue profile
2. Customize the scheduling algorithm and weight (depending on algorithm) for each queue
3. Apply the override schedule profile to the desired interface(s)

Creating the interface override schedule profile called `schedule-profile2`:

```
switch (config)# qos schedule-profile schedule-profile2
switch (config-schedule)# dwrr queue 0 weight 5
switch (config-schedule)# dwrr queue 1 weight 50
switch (config-schedule)# dwrr queue 2 weight 20
switch (config-schedule)# dwrr queue 3 weight 15
switch (config-schedule)# dwrr queue 4 weight 10
```

Applying schedule profile `schedule-profile2` mapped with queue profile `profile1` to the global config:

```
switch(config)# apply qos queue-profile profile1 schedule-profile schedule-profile2
```

Verifying that that the interface override has occurred:

```
switch# show qos schedule-profile
profile_status profile_name
-----
complete      factory-default
applied      schedule-profile2
complete      strict
incomplete    test1
```

Validating that schedule profile `schedule-profile2` is applied to an interface:

```
switch# show interface 1/1/1 qos
Interface 1/1/1 is up
Admin state is up
qos trust none (global)
qos queue-profile profile1 (global)
qos schedule-profile schedule-profile2 (global)
```

Creating the schedule profile called `profile1` that will become the global schedule profile:

```
switch (config)# qos schedule-profile schedule-profile1
switch (config-schedule)# dwrr queue 0 weight 5
switch (config-schedule)# dwrr queue 1 weight 25
switch (config-schedule)# dwrr queue 2 weight 35
switch (config-schedule)# dwrr queue 3 weight 25
switch (config-schedule)# dwrr queue 4 weight 10
```

Applying schedule profile `profile1` to queue profile `profile1`:

```
switch (config)# apply qos queue-profile profile1 schedule-profile schedule-profile1
```

Validating that the schedule profile `profile1` is applied globally and has overwritten the global setting of schedule profile `profile2`:

```
switch# show qos schedule-profile
profile_status profile_name
```

```

-----
complete      factory-default
applied      schedule-profile1
complete      schedule-profile2
complete      strict

```

Validating the global the schedule profile `profile1` has been applied to all interfaces:

```

switch# show interface qos
Interface 1/1/1 is up
Admin state is up
qos trust none (global)
qos queue-profile profile1 (global)
qos schedule-profile schedule-profile1 (global)

output removed for brevity

Interface 1/1/56 is up
Admin state is up
qos trust none (global)
qos queue-profile profile1 (global)
qos schedule-profile schedule-profile1 (global)

```

Applying schedule profile `profile2` to a specific interface to override the global schedule profile:

```

switch (config)# interface 1/1/1
switch (config-if)# apply qos schedule-profile schedule-profile2

```

Verifying that both schedule profiles are applied:

```

switch# show qos schedule-profile
profile_status profile_name
-----
complete      factory-default
applied      schedule-profile1
applied      schedule-profile2
complete      strict

```

Verifying that schedule profile `profile2` has overridden `profile1` on interface `1/1/1` and `profile1` has been applied to all other interfaces:

```

switch# show interface qos
Interface 1/1/1 is up
Admin state is up
qos trust none (global)
qos queue-profile profile1 (global)
qos schedule-profile schedule-profile2 (override)

interfaces removed for brevity

Interface 1/1/56 is up
Admin state is up
qos trust none (global)
qos queue-profile profile1 (global)
qos schedule-profile schedule-profile1 (global)

```

Schedule profile `profile2` has been applied only to interface 1/1/1. All other interfaces retain the global schedule profile setting of `profile1`.

IP explicit congestion notification

TCP/IP networks signal congestion by dropping packets. When packets are dropped due to congestion, TCP/IP easily detects this but performance suffers. For lossless Ethernet fabric, dropping packets is not allowed, so a different action is required to inform endpoints of congestion. Instead of dropping packets when congestion is experienced, ECN marks the IP packet as congestion experienced (CE).

ECN is used in conjunction with DCBx and used to avoid hitting the PFC limits over L3 network hops. The ECN bits of a packet are conditionally marked by a switch based on a configurable ECN threshold which allows TCP to function normally without dropping IP packets. ECN has the following characteristics:

- Uses the DiffServ field in the IP header to mark the congestion status along the packet transmission path.
- End-to-end communication protocol which requires support by each end device as well as all intermediate switch hops.
- Applied on a queue basis and can be applied to 1 or more queues concurrently up to the 8 queue limit.

ECN uses the two "least significant" bits in the Traffic Class (DiffServ) field of the IP header (the bits that are farthest to the right) to signal the current state of network congestion. The possible markings are:

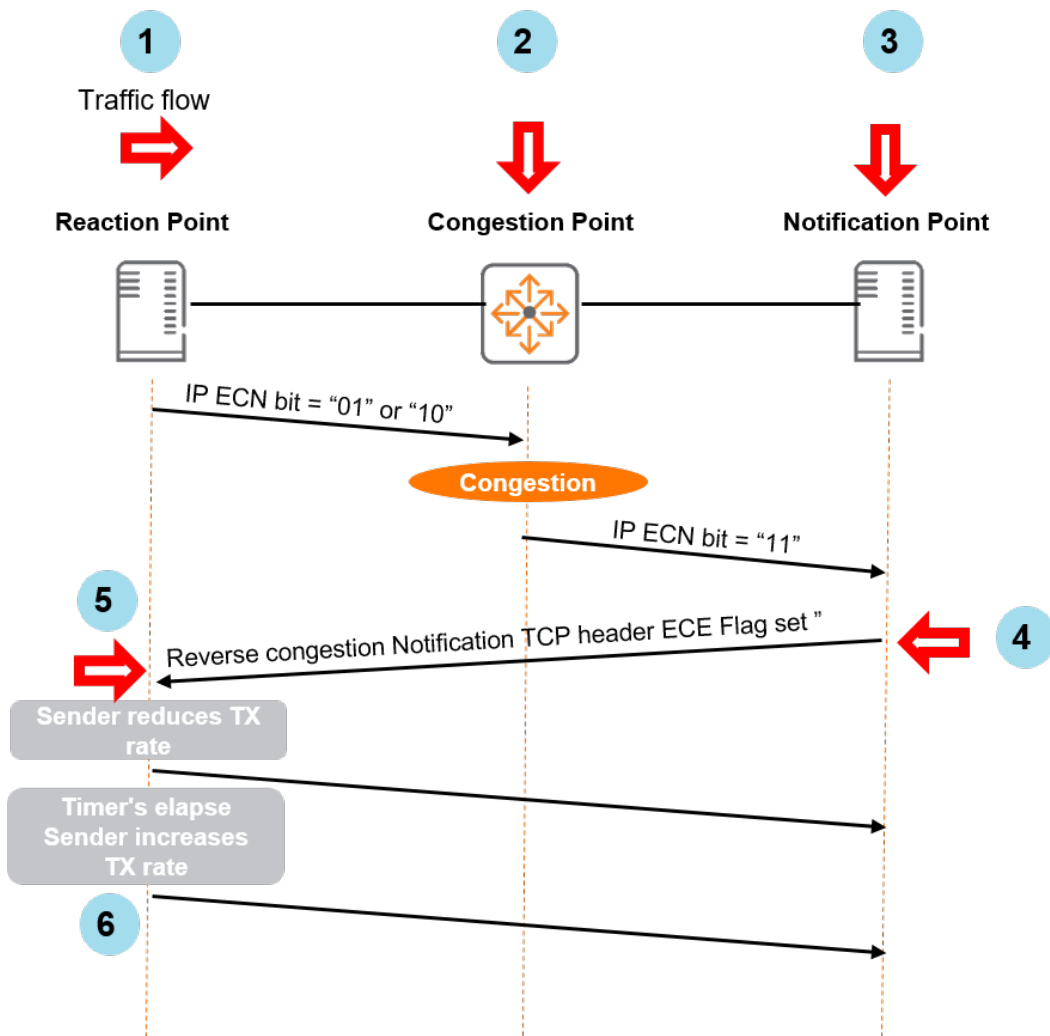
Bits	Congestion status	Abbreviation
00	Non ECN-Capable Transport	Non-ECT
01	ECN Capable Transport	ECT(1)
10	ECN Capable Transport	ECT(0)
11	Congestion Encountered	CE



ECT(1) and ECT(0) are considered to be functionally the same by the switch. If congestion is encountered, the zero bit will be flipped to indicate congestion.

The benefit of ECN is that it reduces the likelihood of dropping packets in a lossy fabric or reduces the usage of flow-control pause in a lossless fabric. ECN accomplishes this via the following actions:

1. IP packets flow normally on an IP network from an ECN-enabled device with the ECN field marked ECT(0) or ECT(1) to indicate support for ECN
2. If an ECN-enabled host device receives a packet marked CE; it acknowledges the marked packet with its upper layer protocols (TCP)
3. The host receiving the CE marked packet sends the congestion notification back to the transmitting host
4. The transmitting host device handles ECE marking in the received acknowledgment frame by reducing its transmit rate
5. The reverse congestion notification occurs in the TCP header, not the IP header, using the ECE flag of the acknowledgment frame.



IP ECN configuration considerations

The HPE Aruba Networking 8400 Switch Series has a slope threshold which is calculated in bytes. When this threshold is met, the packets are immediately marked for ECN congestion. There is no upper threshold limit. Because the packets are marked for congestion when the threshold is met, setting the congestion threshold too low will impair network performance as the end host's TCP flow backs off when it receives the ECN congestion marked bits in the IP packet. Conversely, setting the IP ECN threshold too high may trigger PFC pause between network switches and end hosts.

The following factors should always be taken into account when configuring IP ECN:

- ECN is configured on queues carrying delay-sensitive traffic.
- Any number of queues maybe configured in a profile.
- An unconfigured queue means no threshold action behavior will occur.
- ECN can be applied globally or on individual interfaces.
- No threshold policies are applied by default.
- All switches in the transmission path between two ECN-enabled endpoints must have ECN enabled and configured for the appropriate queue.

- General guidance is to ensure that the ECN threshold is not configured too low or too high:
 - Too low and ECN will impact throughput
 - Too high and ECN will not be triggered and PFC pause will be invoked
- Testing is recommended to ensure ECN thresholds are correctly applied.

IP ECN configuration procedure

The procedure for configuring IP ECN is as follows:

- Create a threshold profile using the [qos threshold-profile](#) command and apply threshold values
- Optionally apply a QoS threshold profile globally
- Optionally apply a QoS threshold profile to the desired interface(s) to override the global threshold profile setting

Creating a QoS threshold profile called **ECN** applied globally to all interfaces:

```
switch(config)# qos threshold-profile ECN
switch(config-threshold)# queue 1 action ecn all threshold 40 kbytes
switch(config)# apply qos threshold-profile ECN
```

Creating a QoS threshold profile called **ECN2** :

```
switch(config)# qos threshold-profile ECN2
switch(config-threshold)# queue 1 action ecn all threshold 100 kbytes
```

Applying QoS threshold profile **ECN2** to a specific interface:

```
switch(config)# interface 1/1/1
switch(config-if)# apply qos threshold-profile ECN2
```

Lossless QoS pool

The QoS pool is a packet buffer pool. The HPE Aruba Networking 8325, 9300, and 10000 Switch Series have a configurable QoS pool with an associated headroom buffer space, while the 8360 has a single fixed QoS pool which is not exposed to the CLI for configuration. The 8400 Switch Series assigns reserved buffer space per-interface per-PFC-priority, there is no fixed lossless pool.

Switch series	Lossless QoS pools
8325/9300/10000	Three configurable lossless pools
8360/8400	Single fixed lossless pool

The QoS pool is a packet buffer pool resident in memory in the switch. Only the QoS pools on the HPE Aruba Networking 8325, 9300, and 10000 Switch Series are configurable and have the following attributes:

- The QoS pool option enables the creation of a packet buffer pool which is dedicated to lossless traffic
- The QoS pool command allows the creation of a lossless pool size, headroom buffer associated with the pool, and the priorities that are mapped to that pool.
- The lossless pool size is a percentage of the total available buffer memory on the device.
- The headroom pool memory is allocated from the lossless pool and is used for storing packets that arrive on a port after a pause has been asserted.

Considerations and prerequisites

There are no prerequisite configuration activities required to configure QoS pools. However, a PFC priority or any related PFC commands cannot be enabled on an interface until the creation of the QoS pool and the appropriate PFC priority mapping to that pool. The following considerations should be taken into account when configuring QoS pools:

- Maximum buffer size in memory
 - 8325/10000: 32Mb
 - 9300: 66 MB
- Maximum configurable headroom memory size: 10240 kbytes (10Mb)
- Factory-default headroom size: 3072 kbytes (3Mb)
- The sum of pool sizes cannot be larger than 90% of the available Kbyte memory
- The headroom size cannot be greater than half of the lossless pool buffer size.



Best practice for QoS pools on all platforms is to not assign the best effort traffic class (default local priority 1) and the critical control traffic (local priority 7) to a lossless pool.

QoS pool configuration

QoS pools are enabled using the QoS pool command in the global configuration context.



The following examples are applicable to the 8325, 9300, and 10000 Switch Series.

Configuration options for QoS pool size:

```
switch(config)# qos pool (1-3) lossless size ?
<10-90>          The percentage of packet buffer memory to be allocated to
                  this pool in integer format
<10.00-90.00>   The percentage of packet buffer memory to be allocated to
                  this pool in decimal format
factory-default  Configure the pool size to the default value
```

Configuration options for headroom size:

```
switch(config)# qos pool 2 lossless size factory-default percent headroom ?
<0-10240>       The headroom buffer size in kilobytes
factory-default Configure the default headroom buffer size in kilobytes
```

Configuring a QoS pool size of 60% of the total available buffer memory and a headroom size of 2048 kB with priorities 1, 2, and 3 mapped to the QoS pool:

```
switch(config)# qos pool 2 lossless size 60 percent headroom 2048 kbytes
priorities 1,2,3
```

The `no` form of the `QoS pool` command will remove the configuration from the switch. For example, removing QoS pool 2:

```
switch (config)# no qos pool 2
```

PFC flow-control priorities must be removed on all interfaces prior to removing the QoS pool otherwise a warning message will be displayed and the QoS pool will not be removed from the configuration. The warning message reads as follows:

```
'flow-control priority' must be unconfigured for all interfaces before deleting
pool 2.
```

Use the `no flow-control priority` command within the appropriate interface context:

```
switch(config-if)# no flow-control priority
<0-7>    The packet priority to flow control
rx       Honor received PAUSE requests for this priority
rxtx    Send and receive PAUSE requests for this priority
tx       Send PAUSE requests for this priority
```

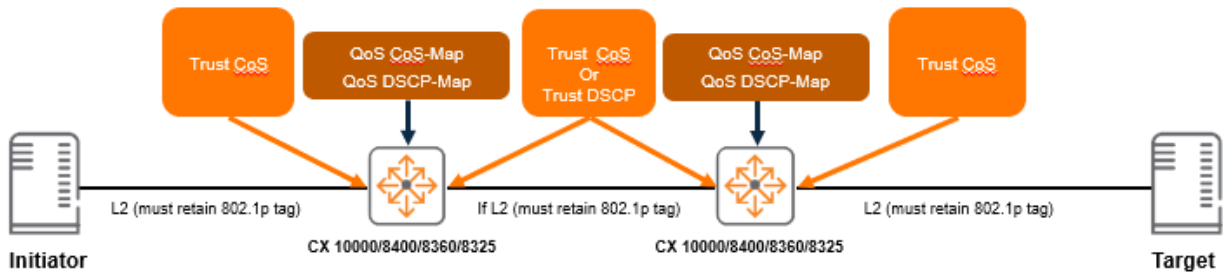
Reapply the `no qos pool` command:

```
switch (config)# no qos pool 2
```

QoS trust

QoS trust can be applied either globally or at the interface level which will override the global setting. The following considerations should be taken into account when configuring QoS trust:

- L2 communications between the switch and the host must leverage the 802.1Q tag meaning the interface between the switch and host must be a trunk link.
- L2 switch fabric 802.1Q tags must be configured between target and initiator across the layer 2 fabric.
- L3 switch fabric must sync the appropriate CoS map to local preferences with the DCSP map local preferences. This is to ensure appropriate mapping for CoS and DSCP markings across the network.



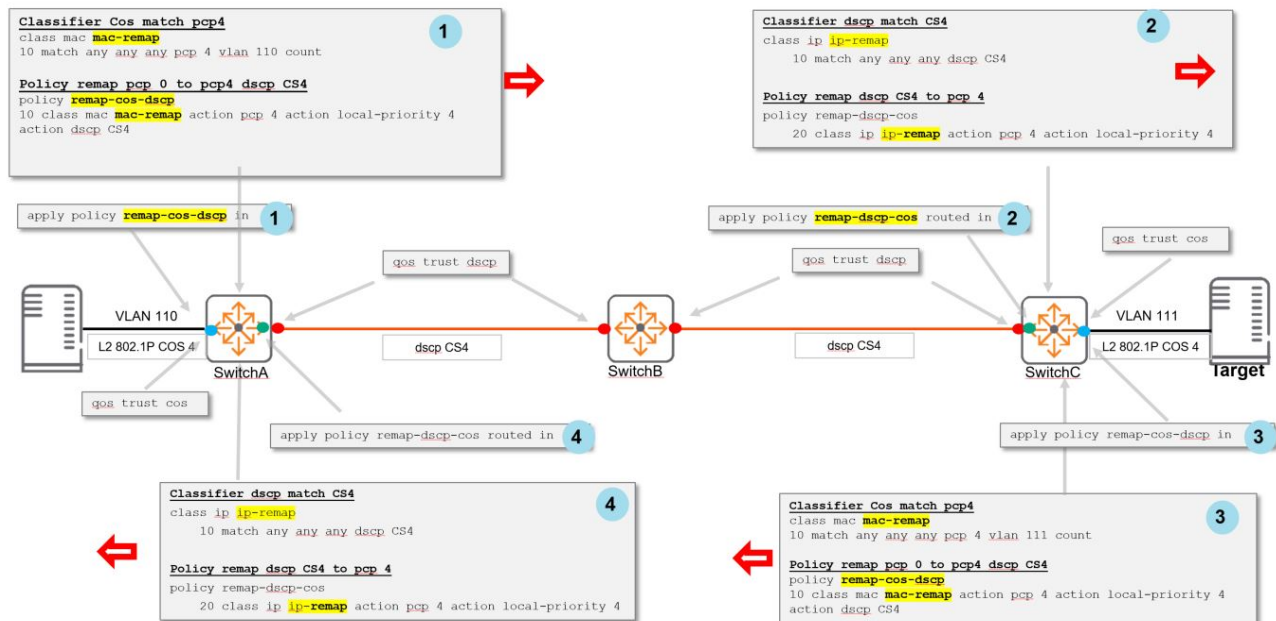
Set a global trust setting or leave global QoS trust the default setting. Override the global trust setting on specific interfaces as desired.



QoS trust is set to none by default.

Layer 3 CoS-DSCP markings

A remark policy may be advantageous in order to ensure that the local priority derived from the "trusted" CoS value is remarked to the desired DSCP code point for transmission across the IP fabric. The diagram illustrates an example of layer 3 switch configuration with remark policy:



Configuration steps:

1. Switch A

- a. A classifier is applied to match traffic with a priority code point (PCP) of 4
- b. A policy is applied to remark the classified traffic to the desired DSCP code point (CS4)
- c. The policy is applied to the switch egress interface

2. Switch C

- a. A classifier is applied to match traffic with a DSCP code point of CS4
- b. A policy is applied to remark the classified traffic to the local-priority of 4 (CoS 4)
- c. The policy is applied to the ingress interface (facing the initiator host)

3. Switch C

- a. A classifier is applied to match traffic with a PCP of 4
- b. A policy is applied to remark the classified traffic to the desired DSCP code point (CS4 in this example)
- c. The policy is applied to the switch egress interface

4. Switch A

- a. A classifier is applied to match traffic with a DSCP code point of CS4
- b. A policy is applied to remark the classified traffic to the local-priority of 4 (CoS 4)
- c. The policy is applied to the routed ingress interface (facing the target host)



The policies applied ensure that the DSCP value is consistent across the L3 network. At the final hop, the DSCP value is mapped to the correct CoS value for transmission to the target host. The policy is applied to the reverse flow.

Verifying trust settings

Global trust verification

```
switch# show qos trust
qos trust cos
```

Interface trust verification

```
switch# show interface 1/1/1 qos
Interface 1/1/1 is up
Admin state is up
qos trust cos (override)
qos queue-profile factory-default (global)
qos schedule-profile factory-default (global)
```

Troubleshooting data center bridging

It is assumed that prior to engaging in troubleshooting that host targets and initiators have network connectivity, are using DCBx, and that hosts are supporting appropriate DCB protocols like PFC. Checks should be applied to every switch hop. The following flow chart represents the HPE Aruba Networking-recommended data center bridging troubleshooting flow supporting common DCB implementation:

1. Check Host Target/Initiators configurations are using 801.q tagged frames to support the 802.1p field.

2. Validate DCBX application, QoS Pools (8325/10000)

3. Check qos-queue profile and qos schedule-profile

4. Check DCBX TLV and PFC configurations

1. Verify that host target/initiator configurations are using 801.Q tagged frames to support the 802.1p field.

Check that both host and initiator are using 802.1Q tagged frames between host and switch. This requires the switch interface to be configured as a trunk interface supporting the appropriate VLANs. Check that the appropriate `qos trust` setting is applied to the interface. This can either globally or via the interface context with the `qos trust` and `show interface 1/1/1 qos` commands. The `qos trust` setting should be set to `qos trust cos` if the host is supporting PFC as well as DCBx and is configured to use the appropriate CoS setting for a specific lossless traffic flow.

2. Validate DCBx application and QoS Pools



Only applicable to the HPE Aruba Networking 8325, 9300, and 10000 Switch Series

Validate that DCBx is configured on the correct interfaces, that the adjacent neighbor is using the correct DCBx version, and using the same PFC priority. If DCBx is configured and the PFC priority matches at either end for the link, the following output is displayed:

```
switch# show dcbx interface 1/1/1
DCBx admin state      : enabled
DCBx operational state : active
DCBx version         : local = ieee, remote = ieee
PFC operational state : active

Mismatch Advertisement      Local Peer
-----
Priority Flow Control (PFC)
Willing:                    No    No
MACsec Bypass Capability:   No    No
Max PFC Traffic Classes:    2    2
Priority 0:                  False False
```

```

Priority 1:           False   False
Priority 2:           False   False
Priority 3:           False   False
Priority 4:           True    True
Priority 5:           False   False
Priority 6:           False   False
Priority 7:           False   False

```

Output omitted for brevity

```

switch(config)# show qos pool config

Global Packet-Buffer Configuration and Status (in Kbytes)

Configured                               Sum of           Applied
Priority                                 Limits           Port           Priority
Packet Buffer Pools Size                 Mapping
Mapping
-----
--
Lossy Pool                               9546             --           0,1,2,5,6,7
0,1,2,5,6,7
Lossless Pool 1                          7546 (30.00%)   --           4
4
Headroom Pool 1                          2000            0            --
--
Lossless Pool 2                          9728 (40.00%)   --           3
3
Headroom Pool 2                          3000            0            --
--
Reserved Memory                          948             --           --
-----
--
TOTAL:                                   32768

```

3. Verify qos-queue profile and qos schedule-profile

Verify that the `qos-queue profile` and the global `qos schedule-profile` have been applied. A single `qos queue-profile` is applied to the system. A schedule is always applied to all interfaces (either the globally configured schedule profile, an interface-override schedule profile, or the factory-default schedule-profile if the configuration cannot be applied). Each port can have its own schedule profile. Verify that the correct QoS queue profile is applied. An example of the QoS queue profile **4qp** applied to the system:

```

switch# show qos queue-profile
profile_status profile_name
-----
complete      3qp
applied       4qp
complete      SMB
complete      factory-default

```

Verify that the correct QoS schedule profile is applied to an interface. An example of the QoS schedule profile **4sp** applied to the system:

```
switch# show interface 1/1/1 qos  
Interface 1/1/1 is up  
Admin state is up  
qos trust cos (override)  
qos queue-profile 4qp (global)  
qos schedule-profile 4sp (global)
```

4. Check DCBx TLV and PFC Configurations

Verify that the correct DCBx application TLV is configured and has the correct priority using the `show dcbx interface` command.

Verify that appropriate interfaces have PFC configured:

```
switch# show interface 1/1/2  
interface 1/1/2  
  no shutdown  
  flow-control priority rxtx 4  
  no routing  
  vlan trunk native 1  
  vlan trunk allowed 1,110  
  qos trust cos
```

Output omitted for brevity

apply qos

```
apply qos [queue-profile <QUEUE-NAME>] schedule-profile <SCHEDULE-NAME>
no apply qos [queue-profile <QUEUE-NAME>] schedule-profile <SCHEDULE-NAME>
```

Description

Applies a queue profile and schedule profile globally to all Ethernet and LAG interfaces on the switch, or applies a schedule profile to a specific interface. When applied globally, the specified schedule profile is configured only on Ethernet interfaces and LAGs that do not already have their own schedule profile.

The same profile can be applied both globally and locally to an interface. This guarantees that an interface always uses the specified profile, even if the global profile is changed.

The **no** form of this command removes the specified schedule profile from an interface and the interface uses the global schedule profile. This is the only way to remove a schedule profile override from the interface.



When applying QoS settings to a port configured to support priority-based flow control, specific configuration settings must be respected when defining a CoS map and queue profile. See the command **flow-control** in the *Command-Line Interface Guide* for details.



Interfaces may shut down briefly during reconfiguration.

Parameter	Description
<code>queue-profile <QUEUE-NAME></code>	Specifies the name of the queue profile to apply. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-). This parameter is not supported in the config-if context.
<code>schedule-profile <SCHEDULE-NAME></code>	Specifies the name of the schedule profile to apply. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Usage

- The switch must always have a globally-applied queue and schedule profile. To stop using a given profile, apply a different profile.
- For a queue profile to be complete and ready to be applied, all eight local priorities must be mapped to a queue.

- For a schedule profile to be complete and ready to be applied, it must define all queues specified in the queue profile. All queues must use the same algorithm, except for the highest numbered queue, which can be **strict**.
- Both the queue profile and the schedule profile must specify the same number of queues.
- Schedule profiles can be modified while applied, but only in ways where a single command will not result in the profile becoming invalid. For example, queue 7 can have the algorithm changed, and weighted queues can have their weights changed.

If there are interfaces running with priority-based flow control (PFC) and a new queue profile to be applied maps a local priority used by the PFC traffic to another queue, all PFC interfaces must be shutdown before applying the new queue profile. If the new queue profile was applied before shutting down the PFC interfaces, PFC traffic will still use the same queue from the previous profile until the interfaces are shutdown then re-enabled.

Examples

The following commands illustrate a valid configuration where every local priority value is assigned to a queue and all assigned queues are defined:

```
switch(config)# qos queue-profile Q1
switch(config)# map queue 0 local-priority 0
switch(config)# map queue 1 local-priority 1
switch(config)# map queue 1 local-priority 2
switch(config)# map queue 3 local-priority 3
switch(config)# map queue 4 local-priority 4
switch(config)# map queue 5 local-priority 5
switch(config)# map queue 5 local-priority 6
switch(config)# map queue 5 local-priority 7
switch(config)# qos schedule-profile S1
switch(config)# wfq queue 0 weight 5
switch(config)# wfq queue 1 weight 10
switch(config)# wfq queue 3 weight 15
switch(config)# wfq queue 4 weight 25
switch(config)# wfq queue 5 weight 50
```

The following commands illustrate an invalid configuration because local priority 2 is not assigned to a queue:

```
switch(config)# qos cos-map 1 local-priority 1
switch(config)# qos queue-profile Q1
switch(config)# map queue 0 local-priority 0
switch(config)# map queue 1 local-priority 1
switch(config)# map queue 3 local-priority 3
switch(config)# map queue 4 local-priority 4
switch(config)# map queue 5 local-priority 5
switch(config)# map queue 5 local-priority 6
switch(config)# map queue 5 local-priority 7
switch(config)# qos schedule-profile S1
switch(config)# wfq queue 0 weight 5
switch(config)# wfq queue 1 weight 10
switch(config)# wfq queue 3 weight 15
switch(config)# wfq queue 4 weight 25
switch(config)# wfq queue 5 weight 50
```

Applying the QoS profile **Q1** and the schedule profile **S1** to all interfaces that do not have an applied interface-specific schedule profile:

```
switch(config)# apply qos queue-profile Q1 schedule-profile S1
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	config config-if config-lag-if	Administrators or local user group members with execution rights for this command.

apply qos threshold-profile

```
apply qos threshold-profile <THRESHOLD-NAME>  
no apply qos threshold-profile
```

Description

Applies a threshold profile globally to all Ethernet and LAG interfaces on the switch, or to a specific interface. When applied globally, the specified threshold profile is configured only on Ethernet interfaces and LAGs that do not already have their own schedule profile.

The same profile can be applied both globally and locally to an interface. This guarantees that an interface always uses the specified threshold profile, even if the global profile is changed.

The **no** form of this command removes the specified threshold profile from an interface, and causes it to use the global threshold profile. This is the only way to remove a threshold profile override from an interface. A profile can only be deleted once it is no longer applied to any interface.

Parameter	Description
<THRESHOLD-NAME>	Specifies the name of the threshold profile to apply. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Example

Applying the threshold profile **mythreshold** to all interfaces that do not have an applied profile:

```
switch(config)# apply qos threshold-profile mythreshold
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config config-if config-lag-if	Administrators or local user group members with execution rights for this command.

dcbx application

```
dcbx application {iscsi | tcp-sctp <PORT-NUM> | tcp-sctp-udp <PORT-NUM> | tcp-udp <PORT-NUM>
                | udp <PORT-NUM> | ether <ETHERTYPE>} priority <PRIORITY>
no dcbx application
```

Description

Configures application to priority map that gets advertised in DCBx application priority messages. This tells the DCBx peer to send the application traffic with the configured priority so that the traffic is treated as lossless. Multiple applications can be configured in this manner. PFC lossless priority configured on the switch should be the same as this priority.

The **no** form of this command removes the existing configuration.

Parameter	Description
iscsi	Specifies a physical port on the switch. TCP ports 860 and 3260.
tcp-sctp <PORT-NUM>	Specifies the traffic for a specified TCP or SCTP port. Range: 1 to 65535.
tcp-sctp-udp <PORT-NUM>	Specifies the traffic for a specified TCP or SCTP or UDP port. Range: 1 to 65535.
tcp-udp <PORT-NUM>	Specifies the traffic for a specified TCP or UDP port. Range: 1 to 65535.
udp <PORT-NUM>	Specifies the traffic for a specified UDP port. Range: 1 to 65535.
<ETHERTYPE>	Specifies the traffic for a specific Ethernet type. Range: 1536 to 65535.
<PRIORITY>	Specifies the application priority. Range: 0 to 7.

Usage

- In CEE DCBx version, the following traffic type configurations are sent using application TLVs:
 - Ethertype
 - iSCSI
 - TCP-UDP
- In IEEE DCBx version, the following traffic type configurations are sent using application TLVs:
 - Ethertype
 - iSCSI
 - TCP-SCTP

- TCP-SCTP-UDP
- UDP

Examples

Mapping iSCSI traffic to priority 5.

```
switch(config)# dcbx application iscsi priority 5
```

Mapping TCP or SCTP traffic with port 860 to priority 3.

```
switch(config)# dcbx application tcp-sctp 860 priority 3
```

Command History

Release	Modification
10.08	Added a parameter option tcp-udp .
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config	Administrators or local user group members with execution rights for this command.

lldp dcbx (global)

```
lldp dcbx [version {cee|ieee}]
no lldp dcbx [version {cee|ieee}]
```

Description

Enables advertisement of the DCBx TLVs in LLDP packets either globally or per interface. By default, DCBx is disabled in the switch.

The **no** form of this command disables DCBx advertisement.

Parameter	Description
version {cee ieee}	Configures the DCBx version in either CEE (Converged Enhanced Ethernet) or IEEE (IEEE 802.1Qaz). Default version: IEEE

Examples

Enabling DCBx globally with default version:

```
switch(config)# lldp dcbx
```

Disabling DCBx globally:

```
switch(config)# no lldp dcbx
```

Enabling DCBx globally with the CEE version:

```
switch(config)# lldp dcbx version cee
```

Enabling DCBx on an interface with default version:

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

Disabling DCBx on an interface:

```
switch(config-if)# no lldp dcbx
```

Enabling DCBx on an interface with the CEE version:

```
switch(config-if)# lldp dcbx version cee
```

Command History

Release	Modification
10.08	Added optional CEE and IEEE parameters.
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config config-if	Administrators or local user group members with execution rights for this command.

lldp dcbx (per interface)

```
lldp dcbx [version {cee|ieee}]  
no lldp dcbx [version {cee|ieee}]
```

Description

Enables DCBx on an interface. By default, an interface follows the global DCBx configuration. DCBx must be enabled globally for the interface configuration to take effect.

The **no** form of this command removes the port-override DCBx configuration and reverts the port behavior to the global DCBx configuration.

Parameter	Description
version { cee ieee }	Configures the DCBx version in either CEE (Converged Enhanced Ethernet) or IEEE (IEEE 802.1Qaz). Default version: IEEE

Usage

If the interface command specifies a different version than the global configuration, it overrides the globally configured DCBx version. If the command is executed without specifying a version, the IEEE version is configured.

Examples

Enabling DCBx on interface 1/1/1:

```
switch(config)# interface 1/1/1
switch(config-if)# lldp dcbx
```

Disabling DCBx on interface 1/1/1:

```
switch(config)# interface 1/1/1
switch(config-if)# no lldp dcbx
```

Enabling DCBx on interface 1/1/1 with the CEE version:

```
switch(config)# interface 1/1/1
switch(config-if)# lldp dcbx version cee
```

Command History

Release	Modification
10.08	Added a parameter 'version'
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400 9300	config-if	Administrators or local user group members with execution rights for this command.

lldp dcbx disable

```
lldp dcbx disable
no lldp dcbx disable
```

Description

Disables DCBx on an interface.

The **no** form of this command removes the DCBx configuration from the interface, which will then default to the global configuration.

Usage

If the interface command specifies a different version than the global configuration, it overrides the globally configured DCBx version. If the command is executed without specifying a version, the IEEE version is configured.

Examples

Disabling DCBx on interface 1/1/1:

```
switch(config)# interface 1/1/1
switch(config-if)# lldp dcbx disable
```

Reverting interface 1/1/1 to the global DCBx configuration:

```
switch(config)# interface 1/1/1
switch(config-if)# no lldp dcbx
```

Command History

Release	Modification
10.08	Command introduced.

Command Information

Platforms	Command context	Authority
8400	config-if	Administrators or local user group members with execution rights for this command.

map queue

```
map queue <QUEUE-NUMBER> local-priority <PRIORITY-NUMBER>
no map queue <QUEUE-NUMBER> [local-priority <PRIORITY-NUMBER>]
```

Description

Assigns a local priority to a queue in a queue profile. By default, the larger the queue number the higher its priority.

The **no** form of this command removes the specified local priority from a specific queue.

Parameter	Description
<QUEUE-NUMBER>	Specifies the queue number. Range: 0 to 7.
<PRIORITY-NUMBER>	Specifies the local priority. Range: 0 to 7, where 0 is the lowest priority and 7 is the highest.

Usage

For a queue profile to be complete and ready to be applied, all eight local priorities must be mapped to a queue. Any local priority used by interface Priority-based Flow Control (PFC) must be the only local priority mapped to its queue. In order for PFC pausing to work as intended, no other local priorities should be mapped to that same queue. This queue mapping should be configured during initial switch provisioning and only changed during maintenance periods where all ports are disabled.

The following commands illustrate a valid configuration, where every local priority value is assigned to a queue:

```
map queue 0 local-priority 0
map queue 1 local-priority 1
map queue 1 local-priority 2
map queue 3 local-priority 3
map queue 4 local-priority 4
map queue 5 local-priority 5
map queue 5 local-priority 6
map queue 5 local-priority 7
```

The following commands illustrate an invalid configuration, because local priority 2 is not assigned to a queue:

```
map queue 0 local-priority 0
map queue 1 local-priority 1
map queue 2 local-priority 3
map queue 3 local-priority 4
map queue 4 local-priority 5
map queue 5 local-priority 6
map queue 5 local-priority 7
```

Examples

Assigning priority **7** to queue **7** in profile **myprofile**:

```
switch(config)# qos queue-profile myprofile
switch(config-queue)# map queue 7 local-priority 7
```

Removing priority **7** from queue **7** in profile **myprofile**:

```
switch(config)# qos queue-profile myprofile
switch(config-queue)# no map queue 7 local-priority 7
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	config-queue	Administrators or local user group members with execution rights for this command.

name queue

```
name queue <QUEUE-NUMBER> <DESCRIPTION>  
no name queue <QUEUE-NUMBER>
```

Description

Assigns a description to a queue in a queue profile. This is for identification purposes and has no effect on configuration.

The **no** form of this command removes the description associated with a queue.

Parameter	Description
<QUEUE-NUMBER>	Specifies the queue number. Range: 0 to 7.
<DESCRIPTION>	Specifies a queue description for identification purposes. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Examples

Assigning the description **priority-traffic** to queue **7**:

```
switch(config)# qos queue-profile myprofile  
switch(config-queue)# name queue 7 priority-traffic
```

Removing the description from queue **7**:

```
switch(config)# qos queue-profile myprofile  
switch(config-queue)# no name queue 7
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	config-queue	Administrators or local user group members with execution rights for this command.

qos cos-map

```
qos cos-map <CODE-POINT> local-priority <PRIORITY-NUMBER> [color <COLOR>] [name  
<DESCRIPTION>]  
no qos cos-map <CODE-POINT>
```

Description

Defines the local priority assigned to incoming packets for a specific 802.1 VLAN priority code point (CoS) value. The CoS map values are used to mark incoming packets when QoS trust mode is set to **cos**. In **trust none** mode, CoS map entry 0 is used to set the port default local priority and color.

To see the default CoS map settings, use the following command:

```
switch# show qos cos-map default
code_point local_priority color name
-----
0          1             green Best_Effort
1          0             green Background
2          2             green Excellent_Effort
3          3             green Critical_Applications
4          4             green Video
5          5             green Voice
6          6             green Internetwork_Control
7          7             green Network_Control
```

The **no** form of this command restores the assignments for a CoS map value to the default setting.

Parameter	Description
<code><CODE-POINT></code>	Specifies an 802.1 VLAN priority CoS value. Range: 0 to 7. Default 0.
<code>local-priority <PRIORITY-NUMBER></code>	Specifies a local priority value to associate with the <code>CODE-POINT</code> value. Range: 0 to 7. Default: 0.
<code>color <COLOR></code>	Reserved for future use.
<code>name <DESCRIPTION></code>	Specifies a description for the CoS setting. The name is for identification only, and has no effect on queue configuration. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Usage

Any code point configured for use by interface Priority-based Flow Control (PFC) must be assigned a unique local priority in the CoS map. No other code point can be assigned that same local priority. This should be configured during initial switch provisioning and only changed during maintenance periods where all ports are disabled.

Examples

Mapping CoS value **1** to a local priority of **2**:

```
switch(config)# qos cos-map 1 local-priority 2
```

Mapping CoS value **1** to the default local priority value:

```
switch(config)# no qos cos-map 1
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config	Administrators or local user group members with execution rights for this command.

qos dscp-map

```
qos dscp-map <CODE-POINT> local-priority <PRIORITY-NUMBER> [color <COLOR>] [name <DESCRIPTION>]
no qos dscp-map <CODE-POINT>
```

Description

Defines the local priority color assigned to incoming packets for a specific IP differentiated services code point (DSCP) value. The DSCP map values are used to prioritize incoming packets when QoS trust mode is set to **dscp**.

The **no** form of this command restores the assignments for a code point to the default setting.

Use **show qos dscp-map** to view the current settings. To see the default DSCP map settings, use the following command:

```
switch# show qos dscp-map default
code_point local_priority color name
-----
0          1             green CS0
1          1             green
2          1             green
3          1             green
4          1             green
5          1             green
...
45         5             green
46         5             green EF
47         5             green
48         6             green CS6
...
61         7             green
62         7             green
63         7             green
```

Parameter	Description
<CODE-POINT>	Specifies an IP differentiated services code point. Range: 0 to 63. Default: 0.
local-priority <PRIORITY-NUMBER>	Specifies a local priority value to associate with the CODE-POINT value. Range: 0 to 7. Default: 0.
color <COLOR>	Configures the QoS CoS map color. The supported colors are

Parameter	Description
	green, red, and yellow. The default color is green.
<code>cos <PCP-VALUE></code>	Specifies an optional 802.1p VLAN Priority Code Point remark value. Range: 0 to 7. Default: No remark.
<code>name <DESCRIPTION></code>	Specifies a description for the DSCP setting. The name is used for identification only, and has no effect on queue configuration. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Examples

Setting code point **1** to a local priority of **2** and a CoS of **0**:

```
switch(config)# qos dscp-map 1 local-priority 2 cos 0
```

Setting code point **1** to the default value:

```
switch(config)# no qos dscp-map 1
```

Setting code point **1, 3-5** to a local priority **2**, and CoS color as **green** with the description **EntryName**:

```
switch(config)# qos dscp-map 1,3-5 local-priority 2 color green name EntryName
```

Command History

Release	Modification
10.13	Added <COLOR> parameters.
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config	Administrators or local user group members with execution rights for this command.

qos queue-profile

```
qos queue-profile <NAME>
no qos queue-profile <NAME>
```

Description

Creates a new QoS queue profile and switches to the **config-queue** context for the profile. Or, if the specified QoS queue profile exists, this command switches to the **config-queue** context for the profile. A queue profile maps queues to local-priority values. Each profile has eight queues numbered 0 to 7. The larger the queue number, the higher its priority during transmission scheduling.

The **no** form of this command removes the specified QoS queue profile. Only profiles that are not currently applied can be removed.

Parameter	Description
<code><NAME></code>	Specifies the name of the QoS queue profile to create or configure. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Examples

Creating the profile **myprofile**:

```
switch(config)# qos queue-profile myprofile
switch(config-queue)#
```

Deleting the profile **myprofile**:

```
switch(config)# no qos queue-profile myprofile
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	config	Administrators or local user group members with execution rights for this command.

qos schedule-profile

```
qos schedule-profile <NAME>
no qos schedule-profile <NAME>
```

Description

Creates a QoS schedule profile and switches to the **config-schedule** context for the profile. If the specified schedule profile exists, this command switches to the **config-schedule** context for the profile. The schedule profile determines the order in which queues are selected to transmit a packet, and the amount of service defined for each queue.

Parameter	Description
<code><NAME></code>	Specifies the name of the QoS queue profile to create or configure. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Usage

Queues in a schedule profile are numbered consecutively starting from zero. Queue zero is the lowest priority queue. The larger the queue number, the higher priority the queue has in scheduling algorithms.

A profile named **factory-default** is defined by default and applied to all interfaces. It cannot be edited or deleted. To see its settings, use the command:

```
switch# show qos schedule-profile factory-default
queue_num algorithm weight
-----
0          wfq         1
1          wfq         1
2          wfq         1
3          wfq         1
4          wfq         1
5          wfq         1
6          wfq         1
7          wfq         1
```

A profile named **strict** is predefined and cannot be edited or deleted. The strict profile services all queues of the queue profile to which it is applied, using the strict priority algorithm.

A schedule profile must be defined on all interfaces at all times.

There are two permitted configurations for a schedule profile:

1. All queues use the same scheduling algorithm (for example, WFQ).
2. The highest queue number uses strict priority, and all remaining (lower) queues use the same algorithm (for example, WFQ). This supports priority scheduling behavior necessary for the IETF RFC 3246 Expedited Forwarding specification (<https://tools.ietf.org/html/rfc3246>).

Only limited changes can be made to an applied schedule profile:

- The weight of a wfq queue.
- The bandwidth of a strict queue or a min-bandwidth queue.
- The algorithm of the strict or WFQ queue can be swapped between WFQ and strict, and vice versa.

Applicable to REST: Any other changes will result in an unusable schedule profile, and the switch will revert to the **factory-default** profile until the profile is corrected.

The **no** form of this command removes the specified QoS schedule profile when it is not applied. Only profiles that are not currently applied to an interface can be removed.

Examples

Creating the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)#
```

Deleting the schedule profile **MySchedule**:

```
switch(config)# no qos schedule-profile MySchedule
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	config	Administrators or local user group members with execution rights for this command.

qos shape

```
qos shape <RATE> [kbps|percent] [burst <SIZE>]
no qos shape
```

Description

Limits the egress bandwidth on an interface to a value that is lower than its line rate. An optional burst value may also be specified.

Errors will be generated in the following events:

- A user configures a port-shaping value that is greater than 100 percent
- A user configures a kbps port-shaping value that is less than the supported minimum kbps value. The supported minimum kbps shaping value can be retrieved using the **show capacities** command.

The **no** form of this command removes shaping from an interface.

Parameter	Description
<RATE>	Specifies the maximum traffic rate in kbps. Range: 1 to 100000000. Alternatively, the bandwidth can also be configured as a percentage of link bandwidth. The supported range is 1-100. Default units are kilobits per second.
<SIZE>	Specifies the maximum burst size in kilobytes. Range: 1 to 64. Default: 16.

Usage

When the traffic rate destined for the port exceeds the configured egress bandwidth, the switch will buffer the excess up to the limit of the queues. Rates larger than the interface's link rate will have no effect. When set on a LAG, each member Ethernet port independently shapes its egress bandwidth to the specified rate.

Examples

Configuring an egress port shaping rate of 400 Mbps on interface **1/1/1** with a burst size of 30 KB:

```
switch(config)# interface 1/1/1
switch(config-if)# qos shape 400000 kbps burst 30
```

Configuring an egress port-shaping rate of 40% on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# qos shape 40 percent
```

Deleting egress port shaping on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# no qos shape
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config-if	Administrators or local user group members with execution rights for this command.

qos threshold-profile

```
qos threshold-profile <NAME>
no qos threshold-profile <NAME>
```

Description

Creates a QoS threshold profile and switches to the **config-threshold** context for the profile. If the specified threshold profile exists, this command switches to the **config-threshold** context for the existing profile. The threshold profile determines the action to take when a threshold is exceeded for each queue.

A threshold profile is composed of up to 8 queues, numbered from 0 to 7. Each queue defines the action to take when buffer utilization exceeds a specific threshold.

Configure queues with the command **queue**.

The **no** form of this command removes the specified QoS threshold profile. Only profiles that are not currently applied to an interface can be removed.

Parameter	Description
<NAME>	Specifies the name of the QoS threshold profile to create or configure. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).

Usage

Queues in a threshold profile can be any valid queue number, although it is also valid to create a threshold profile with no queues specified. Queue zero is the minimum allowed queue number. The maximum allowed queue number may vary by product. For products supporting eight queues, the largest queue number is seven. If an applied threshold profile specifies configuration of a queue

number that is not in use based on the configured queue profile, the threshold configuration of that unused queue is ignored.

Examples

Creating the threshold profile **mythreshold**:

```
switch(config)# qos threshold-profile mythreshold
switch(config-threshold)#
```

Deleting the threshold profile **MySchedule**:

```
switch(config)# no qos threshold-profile mythreshold
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config	Administrators or local user group members with execution rights for this command.

qos trust

```
qos trust none [local-priority <PRIORITY-NUMBER>] [color <COLOR>]
no qos trust
```

Description

In the **config** context:

- This command sets the trust mode that is globally applied to all interfaces that do not have a trust mode configured.
- The **no** form of this command restores all interfaces that do not currently have a trust mode configured to the default setting.

In the **config-if** context:

- This command sets the trust mode override for a specific interface.
- The **no** form of this command clears a trust mode override. The interface then uses the global setting. This is the only way to remove a trust mode override.

Parameter	Description
none	Ignores all packet headers. Ingress packets are assigned the local priority and color values configured for CoS map entry 0. Default.

Example

Setting the global trust mode to **dscp**, which is applied to all interfaces that do not already have an individual trust mode configured. An override is then applied to interface **2/2/2**, and LAG 100, setting trust mode to **cos**:

```
switch(config)# qos trust dscp
switch(config)# interface 2/2/2
switch(config-if)# qos trust cos
switch(config-if)# interface lag 100
switch(config-if)# qos trust cos
```



Note: The warning message *QoS port remark configurations are not applied when the QoS trust mode is cos or dscp*, is seen if a port trust command other than *trust none* is attempted when there is already a remark configuration on the port. To restore the old remark configuration, configure the port trust mode to *none*.



Note: The warning message *QoS port remark configurations are not applied when the global QoS trust mode is cos or dscp*, is seen if a port *no qos trust* command is attempted when there is already a remark configuration on the port and the global trust mode is not *none*. To re-apply the remark configuration, set the port trust mode to *none*.

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	config config-if config-lag-if	Administrators or local user group members with execution rights for this command.

queue action

WRED

HPE Aruba Networking 8400 Switch Series

```
queue <0-7> action wred { green | yellow | red } min-threshold <WRED-MIN-LIMIT> kbytes
max-threshold <WRED-MAX-LIMIT> kbytes max-prob <WRED-MAX-PROB> percent
```

ECN

HPE Aruba Networking 8400 Switch Series

```
queue <0-7> action ecn all threshold <LIMIT> kbytes
no queue <0-7>
```

Description

Defines the threshold settings and action for a specified queue in a threshold-profile. For ECN, when queue utilization exceeds the threshold value, ECT (ECN-Capable Transport) packets will be CE (Congestion Encountered) marked when transmitted.

For WRED, when queue utilization exceeds the threshold value, WRED action will randomly early-drop packets to signal congestion. More than one WRED action can be configured on a single queue for different packet colors.

The **no** form of this command removes the settings for a queue.

Parameter	Description
<0-7>	Specifies the queue number. Range: 0 to 7.
<LIMIT>	Specifies the threshold value in kilobytes. Range: 0 to 1700 kilobytes.
<WRED-MIN-LIMIT>	Specifies the queue minimum utilization threshold value for WRED to probabilistically start dropping packets.
<WRED-MAX-LIMIT>	Specifies the queue maximum utilization threshold value for WRED, after which every packet is dropped.
<WRED-MAX-PROB>	Specifies the maximum WRED probability of dropping a packet for the specified queue. NOTE: Applicable only on the HPE Aruba Networking 6400v2, 8100, 8360v2, 8320, 8325, 8400, 9300/9300S, and 10000 Switch Series.

Examples

Assigning a threshold to queue **7** in profile **mythreshold**:

```
switch(config)# qos threshold-profile mythreshold
switch(config-threshold)# queue 7 action ecn all threshold 4000 kbytes
```

Configuring WRED action on queue 2 for red-, yellow-, and green- colored packets:

```
switch(config)# qos threshold-profile threshprofile
switch(config-threshold)# queue 2 action wred green min-threshold 900 kbytes max-
threshold 1600 kbytes max-prob 70 percent
switch(config-threshold)# queue 2 action wred yellow min-threshold 500 kbytes max-
threshold 1000 kbytes max-prob 82 percent
switch(config-threshold)# queue 2 action wred red min-threshold 50 kbytes max-
threshold 700 kbytes max-prob 95 percent
```

Removing a threshold from queue **7** in profile **mythreshold**:

```
switch(config)# qos threshold-profile mythreshold
switch(config-threshold)# no queue 7
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config-threshold	Administrators or local user group members with execution rights for this command.

rate-limit

```
rate-limit {broadcast|multicast|unknown-unicast} <RATE> {kbps|percent|pps}
no rate-limit {broadcast | multicast | unknown-unicast}
```

Description

Sets the amount of traffic of a specific type that can ingress on an Ethernet port, or on each port of a LAG interface. Rate limits are enforced separately on each individual member of a LAG, not on the LAG as a whole.

The **no** form of this command removes the traffic limit for the specified traffic type.

Parameter	Description
{broadcast multicast unknown-unicast}	Specifies the type of ingress traffic to which the rate limit applies: broadcast , multicast , or unknown-unicast . The multicast rate limit affects multicast and broadcast traffic. The broadcast rate limit only affects broadcast traffic. When both types are applied to the same interface, broadcast packets are limited to the lower of the two rate values. Layer 2 BPDU packets, like spanning tree, are also included in the multicast rate limit.
<RATE> {kbps percent pps}	Specifies the rate limit in kilobits per second, packets per second, or as a percentage of link bandwidth. Range: 22 to 100000000 kbps (in steps of 22 kbps), 43 to 209090910 pps (in steps of 43 pps), or 1-100 percent. The actual rate limit will be approximately equivalent to the minimum of the two step values that are closest to the configured rate (or for percent-mode, the kbps-converted rate). The actual applied rate limit can be verified using the show interface <IF-NAME> qos command. For percentage mode, rate-limits may be shown as "not applied" until after link-up has occurred on the configured port or LAG.

Examples

Limiting **broadcast** traffic to **500kbps** on interface **1/1/3**:

```
switch(config)# interface 1/1/3
switch(config-if)# rate-limit broadcast 1 percent
```

Limiting **multicast** traffic to **4000pps** on interface **1/1/3**:

```
switch(config)# interface 1/1/3
switch(config-if)# rate-limit multicast 4000 pps
```

Limiting **unknown unicast** traffic to **100kbps** on interface **1/1/3**:

```
switch(config)# interface 1/1/3
switch(config-if)# rate-limit unknown-unicast 100 kbps
```

Viewing the results of the previous configuration settings:

```
switch# show interface 1/1/3 qos
Interface 1/1/3 is down (Administratively down)
Admin state is down
Hardware: Ethernet, MAC Address: 1c:98:ec:e3:6a:00
MTU 1500
Full-duplex
rate-limit unknown-unicast 100 kbps (109 actual)
rate-limit broadcast 1 percent (99844 actual)
rate-limit multicast 4000 pps (3990 actual)

Speed 0 Mb/s
Auto-Negotiation is turned on
Input flow-control is off, output flow-control is off
RX
    0 input packets          0 bytes
    0 input error           0 dropped
    0 CRC/FCS
L3:
    ucast: 0 packets, 0 bytes
    mcast: 0 packets, 0 bytes
TX
    0 output packets        0 bytes
    0 input error           0 dropped
    0 collision
L3:
    ucast: 0 packets, 0 bytes
    mcast: 0 packets, 0 bytes
```

Limiting **broadcast** traffic to **50kbps** on LAG **100**:

```
switch# config
switch(config)# interface 1/1/3
switch(config)# interface lag 100
switch(config-if)# rate-limit broadcast 50 kbps
```

Configuring a multicast rate-limit as a percentage of link bandwidth:

```
switch(config)# interface 1/1/3
switch(config-if)# rate-limit multicast 1 percent
```

Configuring an unknown-unicast rate-limit in packets per second:

```
switch(config)# interface 1/1/4
```

```
switch(config-if)# rate-limit unknown-unicast 100 pps
```

Command History

Release	Modification
10.13	Added the percent parameter.
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config-if	Administrators or local user group members with execution rights for this command.

show dcbx interface

```
show dcbx interface <IFNAME> [peer | vsx-peer]
```

Description

Shows the current DCBx status and the configuration of PFC, ETS, and application priority applied on the interface and the status of the TLVs received from the peer.

Parameter	Description
interface <IFNAME>	Specifies the interface name.
peer	Shows peer DCBx information.
vsx-peer	Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

Examples

Showing DCBx on interface 1/1/1 with default DCBx version:

```
switch# show dcbx interface 1/1/1
DCBx admin state      : enabled
DCBx operational state : active
DCBx version          : local = IEEE, remote = IEEE
PFC operational state : active

Mismatch Advertisement      Local      Peer
-----
Priority Flow Control (PFC)
-> Willing:                 No        Yes
    MACsec Bypass Capability: Yes       Yes
    Max PFC Traffic Classes: 1           1
    Priority 0:              False       False
```

```

Priority 1:                False          False
Priority 2:                False          False
Priority 3:                False          False
Priority 4:                True           True
Priority 5:                False          False
Priority 6:                False          False
Priority 7:                False          False

Enhanced Transmission Selection (ETS)
-> Willing:                No             Yes
Credit-Based Shaper:      No             No
Max Traffic Classes:      8             8
Priority 0:                Class 1         Class 1
Priority 1:                Class 0         Class 0
Priority 2:                Class 2         Class 2
Priority 3:                Class 3         Class 3
Priority 4:                Class 4         Class 4
Priority 5:                Class 5         Class 5
Priority 6:                Class 6         Class 6
Priority 7:                Class 7         Class 7
Class 0:                  ETS 5          ETS 5
Class 1:                  ETS 30         ETS 30
Class 2:                  ETS 10         ETS 10
-> Class 3:                ETS 10         ETS 25
-> Class 4:                ETS 25         ETS 10
Class 5:                  ETS 10         ETS 10
Class 6:                  ETS 10         ETS 10
Class 7:                  Strict         Strict

Application Priority Map:
Mismatch Protocol        Protocol ID      Local      Peer
                   Priority      Priority
-----
-> iscsi
tcp-sctp              1001 (0x03E9)  1          1
-> tcp-sctp              1002 (0x03EA)  2          7
EtherType             2000 (0x07D0)  6          6

```

Showing DCBx on interface 1/1/1 with CEE version:

```

switch# show dcbx interface 1/1/1
DCBx admin state      : enabled
DCBx operational state : active
DCBx version          : local = CEE, remote = CEE
PFC operational state : active

Mismatch Advertisement      Local      Peer
-----
Control
  Operating Version:        0          0
  Max Version:              0          0
  Sequence Number:         1          1
-> Acknowledgement Number: 1          0

Priority Flow Control (PFC)
  Operating Version:        0          0
  Max Version:              0          0
  Enabled:                  Yes         Yes
-> Willing:                 No          Yes
  Error:                    No          No
  Max PFC Traffic Classes:  8          8

```

```

Priority 0:           False           False
Priority 1:           False           False
Priority 2:           False           False
Priority 3:           False           False
Priority 4:           True            True
Priority 5:           False           False
Priority 6:           False           False
Priority 7:           False           False

```

Priority Group

```

Operating Version:   0                0
Max Version:         0                0
-> Enabled:          Yes              No
-> Willing:           No              Yes
Error:               No              No
Max Traffic Classes: 8                8
Priority 0:           PGID 1          PGID 1
Priority 1:           PGID 0          PGID 0
Priority 2:           PGID 2          PGID 2
Priority 3:           PGID 3          PGID 3
Priority 4:           PGID 4          PGID 4
Priority 5:           PGID 5          PGID 5
Priority 6:           PGID 6          PGID 6
Priority 7:           PGID 7          PGID 7
-> PG0 Percentage:   5                10
-> PG1 Percentage:   10             25
PG2 Percentage:      30             30
PG3 Percentage:      30             30
PG4 Percentage:      30             30
PG5 Percentage:      30             30
PG6 Percentage:      30             30
PG7 Percentage:      30             30

```

Application Protocol

```

Operating Version:   0                0
Max Version:         0                0
Enabled:             Yes              Yes
-> Willing:           No              Yes
Error:               No              No

```

Application Priority Map:

Mismatch	Protocol	Protocol ID	Local Priority	Peer Priority
->	iscsi			5
	tcp/udp	1001 (0x03E9)	1	1
->	tcp/udp	1002 (0x03EA)	2	7
	EtherType	2000 (0x07D0)	6	6

Showing DCBx on interface 1/1/1 with mismatched version:

```

switch# show dcbx interface 1/1/1
DCBx admin state      : enabled
DCBx operational state : version_mismatch
DCBx version          : local = IEEE, remote = CEE
PFC operational state : active

```

Showing DCBx peer connected to interface 1/1/1 using CEE version:

```

switch# show dcbx interface 1/1/1 peer
DCBx version: CEE
Control
  Operating Version      : 0
  Max Version           : 0
  Sequence Number       : 1
  Acknowledgement Number : 1

Priority Flow Control (PFC)
  Operating Version      : 0
  Max Version           : 0
  Enabled                : Yes
  Willing                : No
  Error                  : No
  Max PFC Traffic Classes: 8
  Priority 0             : False
  Priority 1             : False
  Priority 2             : False
  Priority 3             : False
  Priority 4             : True
  Priority 5             : False
  Priority 6             : False
  Priority 7             : False

Priority Group
  Operating Version      : 0
  Max Version           : 0
  Enabled                : No
  Willing                : Yes
  Error                  : No
  Max Traffic Classes    : 8
  Priority 0             : PGID 1
  Priority 1             : PGID 0
  Priority 2             : PGID 2
  Priority 3             : PGID 3
  Priority 4             : PGID 4
  Priority 5             : PGID 5
  Priority 6             : PGID 6
  Priority 7             : PGID 7
  PG0 Percentage         : 25
  PG1 Percentage         : 25
  PG2 Percentage         : 10
  PG3 Percentage         : 10
  PG4 Percentage         : 5
  PG5 Percentage         : 5
  PG6 Percentage         : 10
  PG7 Percentage         : 10

Application Protocol
  Operating Version      : 0
  Max Version           : 255
  Enabled                : Yes
  Willing                : No
  Error                  : No

Application Priority Map:
  Protocol      Protocol ID      Priority
  -----
  iscsi         1001 (0x03E9)      5
  tcp/udp       1002 (0x03EA)      7
  EtherType     2000 (0x07D0)      6

```

Command History

Release	Modification
10.08	Updated the output to show the DCBx version enhancement.
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show interface queues

show interface <INTERFACE-NAME> queues

Description

Displays interface-level queue statistics.

Parameter	Description
<INTERFACE-NAME>	Specifies the name of an Ethernet port or LAG on the switch. Format: member/slot/port or lag number .

Usage

Statistics include:

- **Tx Bytes:** Total bytes transmitted. The byte count may include packet headers and internal metadata that are removed before the packet is transmitted. Packet headers added when the packet is transmitted may not be included.
- **Tx Packets:** Total packets transmitted.
- **Tx Drops:** The sum of packets that were dropped across all line modules by Virtual Output Queues (VOQs) destined for the egress port queue due to either insufficient capacity, or an ingress Classifier Policy on another port dropping destined for this port. As the counts are read separately from each line packet's module, the sum is not an instantaneous snapshot.
- **Tx Byte Depth:** Largest byte depth (or high watermark) found on any ingress line module Virtual Output Queue (VOQ) destined for the egress port.

Examples

Showing queue statistics for interface **1/1/5**:

```
switch# show interface 1/1/5 queues
Interface 1/1/5 is up
Admin state is up
      Tx Bytes      Tx Packets      Tx Drops      Tx Byte Depth
Q0      157113373520    1890863919         0           1362
Q1      233312143017    2808451320        18          65550
Q2      156814056423    1887257650         0           1392
```

Q3	157441358980	1894815504	0	1374
Q4	157700809294	1897941370	0	1362
Q5	157872849381	1900014146	0	1392
Q6	183486049854	2208268429	0	4398
Q7	231607534141	2787913734	0	65544

Showing queue statistics for interface **lag 1**:

```
switch# show interface lag 1 queues
Aggregate-name lag1
Aggregated-interfaces :
1/1/21 1/1/22
Speed 20000 Mb/s
      Tx Bytes      Tx Packets      Tx Drops      Tx Byte Depth
Q0              0              0              0              0
Q1              0              0              0              0
Q2              0              0              0              0
Q3              0              0              0              0
Q4              0              0              0              0
Q5              0              0              0              0
Q6              0              0              0              0
Q7             3450             25              0             151
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show interface qos

show interface <INTERFACE-NAME> qos

Description

Shows various QoS settings for a specific interface.

Parameter	Description
<INTERFACE-NAME>	Specifies the name of an interface on the switch. Format: member/slot/port or lag number .

Examples

Showing QoS settings for interface **1/1/5**:

```
switch# show interface 1/1/5 qos
Interface 1/1/5 is down
Admin state is up
qos trust none (global)
qos queue-profile factory-default (global)
qos schedule-profile EQSExample (override)
qos threshold-profile threshprofile (global)
rate-limit unknown-unicast 50 pps (41 actual)
rate-limit broadcast 500 kbps (505 actual)
rate-limit multicast 1 percent (10000 actual)
qos shape 200000 kbps (199999 kbps actual) burst 20 (20 actual)
```

Queue	Maximum Bandwidth	Bandwidth Units	Burst (KB)
Q1	10082461	kbps	20
Q4	20164923	kbps	31
Q7	30247384	kbps	15

Showing QoS settings for a two-member lag:

```
switch# show interface lag1 qos
Aggregate-name lag1
Admin state is up
qos trust none (global)
qos queue-profile factory-default (global)
qos schedule-profile EQSExample (override)
qos threshold-profile threshprofile (global)
rate-limit unknown-unicast 50 pps (41 actual)
rate-limit broadcast 500 kbps (505 actual)
rate-limit multicast 1 percent (10000 actual)
qos shape 200000 kbps (199999 kbps actual per interface, 399998 kbps total for LAG) burst 20 (20 actual per interface)
```

Queue	Maximum Bandwidth	Bandwidth Units	Burst (KB)
Q1	10082461	kbps	20
Q4	20164923	kbps	31
Q7	30247384	kbps	15

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show qos cos-map

```
show qos cos-map [default] [vsx-peer]
```

Description

Shows the global QoS CoS code point settings, or the factory default settings.

Parameter	Description
default	Shows the factory default CoS code point settings.
vsx-peer	Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

Examples

Showing the current CoS map:

```
switch# show qos cos-map
code_point local_priority color name
-----
0          2              green Best_Effort
1          0              green Background
2          1              green Spare
3          3              green Excellent_Effort
4          4              green Controlled_Load
5          5              green Video
6          6              green Voice
7          7              green Network_Control
```

Showing the default CoS map:

```
switch# show qos cos-map default
code_point local_priority color name
-----
0          1              green Best_Effort
1          0              green Background
2          2              green Excellent_Effort
3          3              green Critical_Applications
4          4              green Video
5          5              green Voice
6          6              green Internetwork_Control
7          7              green Network_Control
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show qos dscp-map

```
show qos dscp-map [default] [vsx-peer]
```

Description

Displays the current or default global QoS dscp-map.

Parameter	Description
default	Shows the factory default DSCP code point settings.
vsx-peer	Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

Examples

Showing the current QoS DSCP map:

```
switch# show qos dscp-map
code_point local_priority color name
-----
0          1              green CS0
1          1              green
2          1              green
3          1              green
4          1              green
5          1              green
6          1              green
7          1              green
8          0              green CS1
...
45         5              green
46         7              green EF
47         5              green
48         6              green CS6
...
61         7              green
62         7              green
63         7              green
```

Showing the default QoS DSCP map:

```
switch# show qos dscp-map default
code_point local_priority color name
-----
0          1              green CS0
1          1              green
2          1              green
```

```

3          1          green
4          1          green
5          1          green
...
45         5          green
46         5          green   EF
47         5          green
48         6          green   CS6
...
61         7          green
62         7          green
63         7          green

```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show qos queue-profile

```
show qos queue-profile [<NAME> | factory-default] [vsx-peer]
```

Description

Shows the status of all queue profiles, or a specific queue profile.

Parameter	Description
<NAME>	Specifies the name of a queue profile. Range 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).
[factory-default]	Specifies the factory default queue profile.
vsx-peer	Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

Usage

The status of a queue profile can be:

- Applied - The profile is actively being used by the switch.
- Complete - The profile meets the criteria to be applied.
- Incomplete - The profile does not meet the criteria to be applied.

For a queue profile to be complete and ready to be applied:

- All eight local priorities must be mapped to some queue.
- There must be 8 queues.

Examples

Showing the settings of the factory default queue profile:

```
switch# show qos queue-profile factory-default
queue_num local_priorities name
-----
0          0
1          1
2          2
3          3
4          4
5          5
6          6
7          7
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show qos schedule-profile

```
show qos schedule-profile [<NAME> | factory-default | strict] [vsx-peer]
```

Description

Shows the status of all schedule profiles, or a specific schedule profile.

Parameter	Description
<NAME>	Specifies the name of a queue or schedule profile. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).
[factory-default]	Specifies the factory default queue profile.
vsx-peer	Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

Usage

The status of a schedule profile can be:

- Applied - The profile is actively being used by one or more ports.
- Complete - The profile meets the criteria to be applied.
- Incomplete - The profile does not meet the criteria to be applied.

For a schedule profile to be complete and ready to be applied it must have:

- An algorithm for each queue defined by the applied queue profile.
- All queues must use the same algorithm except for the highest numbered queue, which may be strict.

Example

Showing the status of all schedule profiles:

```
switch# show qos schedule-profile
profile_status profile_name
-----
applied         MySchedule
complete       factory-default
complete       Test
```

Showing the configuration of factory default schedule profile:

```
switch# show qos schedule-profile factory-default
Queue
Number  Algorithm      Weight
-----
0       wfq            1
1       wfq            1
2       wfq            1
3       wfq            1
4       wfq            1
5       wfq            1
6       wfq            1
7       wfq            1
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show qos threshold-profile

```
show qos threshold-profile [<NAME>] [vsx-peer]
```

Description

Shows the status of all threshold profiles, or a specific threshold profile.

Parameter	Description
<NAME>	Specifies the name of a threshold profile. Range: 1 to 64 alphanumeric characters, including period (.), underscore (_), and hyphen (-).
vsx-peer	Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

Usage

Status fields are:

- applied: The threshold profile is applied to all configured ports.
- partially applied: The threshold profile is applied to some configured ports but failed on other configured ports.
- not applied: The threshold profile could not be applied to any configured ports.
- blank field: The threshold profile is not applied in the configuration (globally or as a port override).
- error: Insufficient TCAM Resources: The switch hardware failed to activate a threshold profile.

Examples

Showing the status of threshold profile **mythreshold**:

```
switch# show qos threshold-profile mythreshold
queue_num action  Threshold
-----
5          ecn      2000
7          ecn      5000

Ports      Status
-----
1/1/1     applied
1/1/8     applied
1/2/2     applied
```

Showing the status of threshold profile **CustomThresh** that failed to be activated by the switch hardware:

```
switch# show qos threshold-profile mythreshold
queue_num action  Threshold
-----
3          ecn      3000
4          ecn      4000

Ports      Status
-----
1/2/2     Error: Insufficient TCAM Resources
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

show qos trust

```
show qos trust [default] [vsx-peer]
```

Description

Shows the global QoS trust settings, or the factory default settings.

Parameter	Description
default	Shows the factory default QoS trust settings.
vsx-peer	Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

Examples

Showing the current QoS trust settings:

```
switch# show qos trust
qos trust cos
```

Showing the default QoS trust settings:

```
switch# show qos trust default
qos trust none
```

Command History

Release	Modification
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	Operator (>) or Manager (#)	Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

strict queue

```
strict queue <0-7> [max-bandwidth <RATE> [kbps|percent]][burst <SIZE>]]
no strict queue <0-7> [max-bandwidth <RATE> [kbps|percent]][burst <SIZE>]]
```

Description

Assigns the strict priority algorithm to a queue. Strict priority services all packets waiting in a queue, before servicing the packets in lower priority queues.

Egress queue shaping can be configured using the **max-bandwidth** and **burst** options to limit the amount of traffic transmitted per output queue. The buffer associated with each egress queue stores the excess traffic to absorb bursts and smooth the output rate. Sustained rates of traffic above the maximum bandwidth will eventually fill the output queue, causing tail drops. Use **show interface <IF-NAME> queues** to determine if any tail-drop errors have occurred.

The **no** form of this command removes the the queue configuration from the schedule profile. To remove only egress queue shaping, re-enter the strict queue command without the **max-bandwidth** and **burst** parameters.

Parameter	Description
<QUEUE-NUMBER>	Specifies the number of the queue. Range: 0 to 7.
max-bandwidth <BANDWIDTH>	Specifies the maximum bandwidth allowed on the queue in Kbps. Range: 468 to 100000000. Alternatively, the maximum bandwidth rate can also be configured on the queue as a percentage of the port shape or link bandwidth if a port shape is not configured. The allowed range is 1-100.
burst <SIZE>	Specifies the maximum burst allowed on the queue. HPE Aruba Networking 8320, 8325 and 10000 Switch Series: Range: 1 to 1073. Default: 32. HPE Aruba Networking 9300 Switch Series: Range: 1 to 1024. Default: 32. HPE Aruba Networking 8400 Switch Series: Range: 1 to 32. Default: 32.

Usage

Either all the queues of the schedule profile can be *strict* or just the highest numbered queue. When applied to a LAG, each member Ethernet port independently schedules its egress transmissions using the strict settings. Only limited changes can be made to a *strict* queue that is part of an applied schedule profile:

- The max-bandwidth settings.
- The highest numbered queue can be swapped between *strict* and *wfq*

Any other changes or removing a queue (**no strict queue**) will result in an unusable schedule profile. If that schedule profile is applied in the interface context, the switch will revert to the schedule profile

applied in the global context until the profile is corrected. If that schedule profile is applied in the global context, the switch will revert to using the factory-default profile until the profile is corrected.

It is possible for the following errors to occur:

- **The max-bandwidth cannot be greater than 100 percent.**

This error occurs when a max-bandwidth value greater than 100 percent is configured on a queue.

- **The max-bandwidth cannot be less than <NUM> kbps.**

This error occurs when a kbps max-bandwidth value less than the supported minimum kbps shape value is configured on a queue. The supported minimum kbps shape value can be retrieved using the **show capacities** command.

Examples

Assigning strict priority to queue **7** in the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# strict queue 7
```

Deleting strict priority from queue **7** in the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# no strict queue 7
```

Assigning strict priority to queue **7** in the schedule profile **MySchedule** with a maximum bandwidth of 10000 Kbps and a burst of 62 Kbytes:

```
switch(config)# qos schedule-profile myschedule
switch(config-schedule)# strict queue 7 max-bandwidth 10000 kbps burst 30
```

Assigning strict priority to queue **7** in the schedule profile **MySchedule** with a maximum bandwidth of 40 percent and a burst of 62 Kbytes:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# strict queue 7 max-bandwidth 40 percent burst 30
```

Command History

Release	Modification
10.13	Added the kbps and percent parameters.
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
All platforms	config-schedule-<NAME>	Administrators or local user group members with execution rights for this command.

wfq queue

```
wfq queue <QUEUE-NUMBER> weight <WEIGHT> [max-bandwidth <RATE> {kbps|percent} [burst <SIZE>]]
no wfq queue <QUEUE-NUMBER> weight <WEIGHT> [max-bandwidth <RATE> {kbps|percent} [burst <SIZE>]]
```

Description

Assigns the weighted fair queuing (WFQ) algorithm and its weight to a queue. Weighted fair queuing allocates available bandwidth among all queues that are not empty, in relation to their queue weights. WFQ is applied in bytes, not packets. It is work conserving, which means that only non-empty queues are counted on each scheduling cycle. The percentage of bandwidth allotted to a non-empty queue is calculated by dividing its weight by the sum of the weights for all non-empty queues. This means that the percentage of bandwidth allotted to a queue can fluctuate, depending on the number of non-empty queues present in each cycle.

Egress queue shaping can be configured using the **max-bandwidth** option to limit the amount of traffic transmitted per output queue at all times, even when there is leftover bandwidth available on the port. The buffer associated with each egress queue stores the excess traffic to smooth the output rate.

Sustained rates of traffic above the maximum bandwidth will eventually fill the output queue, causing tail drops. Use **show interface <IF-NAME> queues** to determine if any tail-drop errors have occurred.

The **no** form of this command removes the weighted fair queuing algorithm from a queue. To remove only egress queue shaping, re-enter the wfq queue command without the **max-bandwidth** parameter.

The following errors may occur:

- **The max-bandwidth cannot be greater than 100 percent.** This error occurs when a max-bandwidth value greater than 100 percent is configured on a queue.
- **The max-bandwidth cannot be less than <NUM> kbps.** This error occurs when a kbps max-bandwidth value less than the supported minimum kbps shape value is configured on a queue. The supported minimum kbps shape value can be retrieved using the **show capacities** command.

Parameter	Description
<QUEUE-NUMBER>	Specifies the queue number. Range: 0 to 7.
weight <WEIGHT>	Specifies the scheduling weight. Range: 1 to 253.
max-bandwidth <RATE> {kbps percent}	Specifies the maximum bandwidth rate allowed on the queue in kbps. Range: 64 to 100000000. Alternatively, the maximum bandwidth rate can be configured on the queue as a percentage of the port shape or link bandwidth if a port shape is not configured. The allowed range is 1-100.
burst <SIZE>	Specifies the maximum burst size allowed on the queue in kbytes. Range: 1 to 32. Default: 32.

Examples

Assigning WFQ with a weight of **17** to queue **2** in the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# wfq queue 2 weight 17
```

Assigning WFQ with a weight of 17 and max-bandwidth of 10000 kbps to queue 2 in the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# wfq queue 2 weight 17 max-bandwidth 10000 kbps
```

Assigning WFQ with a weight of 17 and max-bandwidth of 10000 kbps and a burst of 32 kbytes to queue 2 in the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# wfq queue 2 weight 17 max-bandwidth 10000 kbps burst 32
```

Assigning WFQ with a weight of 17 and max-bandwidth of 80 percent and a burst of 32 kbytes to queue 2 in the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# wfq queue 2 weight 17 max-bandwidth 80 percent burst 32
```

Deleting WFQ for queue 2 from the schedule profile **MySchedule**:

```
switch(config)# qos schedule-profile MySchedule
switch(config-schedule)# no wfq queue 2
```

Command History

Release	Modification
10.13	Added the max-bandwidth and burst parameters.
10.07 or earlier	--

Command Information

Platforms	Command context	Authority
8400	config-schedule-<NAME>	Administrators or local user group members with execution rights for this command.

Accessing HPE Aruba Networking Support

HPE Aruba Networking Support Services	https://www.hpe.com/us/en/networking/hpe-aruba-networking-support-services.html
AOS-CX Switch Software Documentation Portal	https://arubanetworking.hpe.com/techdocs/AOS-CX/help_portal/Content/home.htm
HPE Aruba Networking Support Portal	https://networkingsupport.hpe.com/home
North America telephone	1-800-943-4526 (US & Canada Toll-Free Number) +1-650-750-0350 (Backup—Toll Number)
International telephone	https://www.hpe.com/psnow/doc/a50011948enw

Be sure to collect the following information before contacting Support:

- Technical support registration number (if applicable)
- Product name, model or version, and serial number
- Operating system name and version
- Firmware version
- Error messages
- Product-specific reports and logs
- Add-on products or components
- Third-party products or components

Other useful sites

Other websites that can be used to find information:

HPE Aruba Networking Developer Hub	https://developer.arubanetworks.com/hpe-aruba-networking-aoscx/docs/about
Airheads social forums and Knowledge Base	https://community.arubanetworks.com/
AOS-CX Software Technical Update channel on YouTube.	Videos on new features introduced in this release: https://www.youtube.com/playlist?list=PLsYGHuNuBZcbWPEjjHuVMqP-Q_UL3CskS

HPE Aruba Networking Hardware Documentation and Translations Portal	https://arubanetworking.hpe.com/techdocs/hardware/DocumentationPortal/Content/home.htm
HPE Aruba Networking software	https://networkingsupport.hpe.com/downloads
Software licensing and Feature Packs	https://licensemanagement.hpe.com/
End-of-Life information	https://networkingsupport.hpe.com/end-of-life

Accessing Updates

You can access updates from the HPE Aruba Networking Support Portal at <https://networkingsupport.hpe.com>.

Some software products provide a mechanism for accessing software updates through the product interface. Review your product documentation to identify the recommended software update method.

To subscribe to eNewsletters and alerts:

<https://networkingsupport.hpe.com/notifications/subscriptions> (requires an active HPE Aruba Networking Support Portal account to manage subscriptions). Security notices are viewable without an HPE Aruba Networking Support Portal account.

Warranty Information

To view warranty information for your product, go to <https://www.arubanetworks.com/support-services/product-warranties/>.

Regulatory Information

To view the regulatory information for your product, view the *Safety and Compliance Information for Server, Storage, Power, Networking, and Rack Products*, available at <https://www.hpe.com/support/Safety-Compliance-EnterpriseProducts>

Additional regulatory information

HPE Aruba Networking is committed to providing our customers with information about the chemical substances in our products as needed to comply with legal requirements, environmental data (company programs, product recycling, energy efficiency), and safety information and compliance data, (RoHS and WEEE). For more information, see <https://www.arubanetworks.com/company/about-us/environmental-citizenship/>.

Documentation Feedback

HPE Aruba Networking is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback

docsfeedback-switching@hpe.com). When submitting your feedback, include the document title, part number, edition, and publication date located on the front cover of the document. For online help content, include the product name, product version, help edition, and publication date located on the legal notices page.