EVC Quality of Service

First Published: March 28, 2011
Last Updated: March 28, 2011

This document contains information about how to enable quality of service (QoS) features (such as traffic classification and traffic policing) for use on an Ethernet virtual circuit (EVC).

An EVC as defined by the Metro Ethernet Forum is a port-level point-to-point or multipoint-to-multipoint circuit. It is an end-to-end representation of a single instance of a service being offered by a provider to a customer. It embodies the different parameters on which the service is being offered.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Configuring EVC Quality of Service" section on page 14.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Information About Quality of Service on an EVC

EVC Quality of Service and the MQC

QoS functionality is typically applied using traffic classes, class maps, and policy maps. For example, you can specify that traffic belonging to a particular class be grouped into specific categories, and receive a specific QoS treatment (such as classification or policing). The QoS treatment the traffic is to receive is specified in a policy map and the policy map is attached to an interface. The mechanism used for applying QoS in this manner is the modular QoS CLI (MQC.)

The policy map can be attached to an interface in either the incoming (ingress) or outgoing (egress) direction with the **service-policy** command.

The MQC structure allows you to define a traffic class, create a traffic policy, and attach the traffic policy to an interface (in this case, an EVC).

The MQC structure consists of the following three high-level steps.

1. Define a traffic class by using the **class-map** command. A traffic class is used to classify traffic.
2. Create a traffic policy by using the **policy-map** command. (The terms *traffic policy* and *policy map* are often synonymous.) A traffic policy (policy map) contains a traffic class and one or more QoS features that will be applied to the traffic class. The QoS features in the traffic policy determine how to treat the classified traffic.
3. Attach the traffic policy (policy map) to the interface by using the **service-policy** command.

Note For more information about the MQC, including information about hierarchical policy maps and class maps, see the "Applying QoS Features Using the MQC" module.

QoS-Aware Ethernet Flow Point (EFP)

As described in the "EVC Quality of Service and the MQC" section on page 2, the MQC is used to apply one or more QoS features to network traffic. The last step in using the MQC is to attach the traffic policy (policy map) to an interface (in this case, an EVC) by using the **service-policy** command.

With the EVC Quality of Service feature, the **service-policy** command can be used to attach the policy map to an Ethernet Flow Point (EFP) in either the incoming (ingress) *or* outgoing (egress) direction of an EVC. This way, the EFP is considered to be "QoS-aware."

QoS Functionality and EVCs

The specific QoS functionality available on an EVC varies by Cisco IOS XE release but can include the following:

- Packet classification (for example, based on differentiated services code point (DSCP) value and QoS group identifier)
- Packet marking (for example, based on Class of Service (CoS) value)
- Traffic policing (two- and three-color and multiple actions)
- Bandwidth sharing
- Priority queueing (in the outbound direction on the EVC only)
- Weighted Random Early Detection (WRED)

The QoS functionality is enabled by using the appropriate commands listed in the following sections.

- match Commands Supported by EVC QoS for Classifying Traffic, page 3
- Commands Used to Enable QoS Features on the EVC, page 4

match Commands Supported by EVC QoS for Classifying Traffic

Table 1 lists *some* of the available **match** commands that can be used when classifying traffic on an EVC. The available **match** commands vary by Cisco IOS XE release. For more information about the commands and command syntax, see the *Cisco IOS Quality of Service Solutions Command Reference.*

Table 1 match Commands That Can Be Used with the MQC

Command	Purpose
match access-group	Configures the match criteria for a class map on the basis of the specified access control list (ACL).
match any	Configures the match criteria for all packets.
match cos	Matches a packet based on a Layer 2 CoS marking.
match cos inner	Matches the inner CoS of QinQ packets on a Layer 2 CoS marking.
match [**ip**] **dscp**	Identifies a specific IP DSCP value as a match criterion. Up to eight DSCP values can be included in one match statement.
match not	Specifies the single match criterion value to use as an unsuccessful match criterion. **Note** The **match not** command, rather than identifying the specific match parameter to use as a match criterion, is used to specify a match criterion that prevents a packet from being classified as a member of the class. For instance, if the **match not qos-group 6** command is issued while you configure the traffic class, QoS group 6 becomes the only QoS group value that is not considered a successful match criterion. All other QoS group values would be successful match criteria.
match [**ip**] **precedence**	Identifies IP precedence values as match criteria.
match qos-group	Identifies a specific QoS group value as a match criterion.

Table 1 match Commands That Can Be Used with the MQC (continued)

Command	Purpose
match source-address mac	Uses the source MAC address as a match criterion. **Note** Classifying traffic using the **match source-address mac** command is supported in the input direction only.
match vlan (QoS)	Matches and classifies traffic on the basis of the VLAN identification number.
match vlan inner	Configures a class map to match the innermost VLAN ID in an 802.1q tagged frame.

Multiple match Commands in One Traffic Class

If the traffic class contains more than one **match** command, you need to specify how to evaluate the **match** commands. You specify this by using either the **match-any** or **match-all** keyword of the **class-map** command. Note the following points about the **match-any** and **match-all** keywords:

- If you specify the **match-any** keyword, the traffic being evaluated by the traffic class must match *one* of the specified criteria.
- If you specify the **match-all** keyword, the traffic being evaluated by the traffic class must match *all* of the specified criteria.
- If you do not specify either keyword, the traffic being evaluated by the traffic class must match *all* of the specified criteria (that is, the behavior of the **match-all** keyword is used).

Commands Used to Enable QoS Features on the EVC

The commands used to enable QoS features vary by Cisco IOS XE release. Table 2 lists *some* of the available commands and the QoS features that they enable. For complete command syntax, see the *Cisco IOS Quality of Service Solutions Command Reference*.

For more information about a specific QoS feature that you want to enable, see the appropriate module of the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Table 2 Commands Used to Enable QoS Features

Command	Purpose
bandwidth	Configures a minimum bandwidth guarantee for a class.
bandwidth remaining	Configures an excess weight for a class.
drop	Discards the packets in the specified traffic class.
fair-queue	Enables the flow-based queueing feature within a traffic class.
police	Configures traffic policing. Allows specifying of multiple policing actions.
police (percent)	Configures traffic policing on the basis of a percentage of bandwidth available on an interface.
police (two rates)	Configures traffic policing using two rates, the committed information rate (CIR) and the peak information rate (PIR).
priority	Gives priority to a class of traffic belonging to a policy map.

Table 2 Commands Used to Enable QoS Features (continued)

Command	Purpose
queue-limit	Specifies or modifies the maximum number of packets the queue can hold for a class configured in a policy map.
random-detect	Enables Weighted Random Early Detection (WRED).
random-detect discard-class	Configures the WRED parameters for a discard-class value for a class in a policy map.
random-detect discard-class-based	Configures WRED on the basis of the discard class value of a packet.
random-detect exponential-weighting-constant	Configures the exponential weight factor for the average queue size calculation for the queue reserved for a class.
random-detect precedence	Configure the WRED parameters for a particular IP Precedence for a class policy in a policy map.
service-policy	Specifies the name of a traffic policy used as a matching criterion (for nesting traffic policies [hierarchical traffic policies] within one another).
set cos	Sets the Layer 2 CoS value of an outgoing packet.
set discard-class	Marks a packet with a discard-class value.
set [**ip**] **dscp**	Marks a packet by setting the DSCP value in the type of service (ToS) byte.
set mpls experimental	Designates the value to which the Multiprotocol Label Switching (MPLS) bits are set if the packets match the specified policy map.
set precedence	Sets the precedence value in the packet header.
set qos-group	Sets a QoS group identifier (ID) that can be used later to classify packets.
shape	Shapes traffic to the indicated bit rate according to the algorithm specified.

input and output Keywords of the service-policy Command

As a general rule, the QoS features configured in the traffic policy can be applied to packets entering the interface or to packets leaving the interface. Therefore, when you use the **service-policy** command, you need to specify the direction of the traffic policy by using the **input** or **output** keyword.

For instance, the **service-policy output policy-map1** command would apply the QoS features in the traffic policy to the interface in the output direction. All packets leaving the interface (output) are evaluated according to the criteria specified in the traffic policy named policy-map1.

Note For Cisco IOX XE Release 2.1 and later releases, queueing mechanisms are not supported in the input direction. Nonqueueing mechanisms (such as traffic policing and traffic marking) are supported in the input direction.

Also, classifying traffic on the basis of the source MAC address (using the **match source-address mac** command) is supported in the input direction only.

How to Configure a Quality of Service Feature on an EVC

Creating a Traffic Class for Use on the EVC

To create a traffic class, use the **class-map** command to specify the traffic class name. Then use one or more **match** commands to specify the appropriate match criteria. Packets matching the criteria that you specify are placed in the traffic class.

To create the traffic class for use on the EVC, complete the following steps.

Note The **match cos** command shown in Step 4 is an example of a **match** command that you can use. For information about the other available **match** commands, see Table 1 on page 3.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** [**match-all** | **match-any**] *class-name*
4. **match cos** *cos-number*
5. Enter additional **match** commands, if applicable; otherwise, continue with Step 6.
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**class-map** [**match-all** | **match-any**] *class-name* **Example:** `Router(config)# class-map match-any class1`	Creates a class map and enters class-map configuration mode. • The class map is used for matching packets to the specified class. **Note** The **match-all** keyword specifies that all match criteria must be met. The **match-any** keyword specifies that one of the match criteria must be met. Use these keywords only if you will be specifying more than one **match** command.
Step 4	**match cos** *cos-number* **Example:** `Router(config-cmap)# match cos 2`	Matches a packet on the basis of a Layer 2 CoS number. **Note** The **match cos** command is an example of a **match** command you can use. For information about the other **match** commands that are available, see Table 1 on page 3.
Step 5	Enter additional **match** commands, if applicable; otherwise, continue with Step 6.	—
Step 6	**end** **Example:** `Router(config-cmap)# end`	(Optional) Exits class map configuration mode and returns to privileged EXEC mode.

Creating a Traffic Policy (Policy Map) for Use on the EVC

To create a traffic policy (or policy map) for use on the EVC, complete the following steps.

Note The **police** command shown in Step 5 is an example of one of the commands that you can use in a policy map. For information about other available commands, see Table 2 on page 4.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** {*class-name* | **class-default**}
5. **police** *bps* [*burst-normal*] [*burst-max*] [**conform-action** *action*] [**exceed-action** *action*] [**violate-action** *action*]
6. Enter the commands for any additional QoS feature that you want to enable on the EVC, as applicable; otherwise, continue with step 7.
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map policy1`	Creates or specifies the name of the traffic policy and enters QoS policy-map configuration mode.
Step 4	**class** {*class-name* | **class-default**} **Example:** `Router(config-pmap)# class class1`	Specifies the name of a class and enters QoS policy-map class configuration mode. • Enter the class name created in the "Creating a Traffic Class for Use on the EVC" section on page 6 **Note** This step associates the traffic class with the traffic policy.
Step 5	**police** *bps* [*burst-normal*] [*burst-max*] [**conform-action** *action*] [**exceed-action** *action*] [**violate-action** *action*] **Example:** `Router(config-pmap-c)# police 3000`	(Optional) Configures traffic policing. **Note** The **police** command is an example of a command that you can use in a policy map to enable a QoS feature. For information about the other commands available, see Table 2 on page 4.
Step 6	Enter the commands for any additional QoS feature that you want to enable, if applicable; otherwise, continue with Step 7.	—
Step 7	**end** **Example:** `Router(config-pmap-c)# end`	(Optional) Exits QoS policy-map class configuration mode and returns to privileged EXEC mode.

Configuring the EVC and Attaching a Traffic Policy to the EVC

The traffic policy (policy map) applies the enabled QoS feature to the traffic class once you attach the policy map to the EVC.

To configure the EVC and attach a traffic policy to the EVC, complete the following steps.

Note One of the commands used to attach the traffic policy to the EVC is the **service-policy** command. When you use this command, you must specify either the **input** or **output** keyword along with the policy map name. The policy map contains the QoS feature you want to use. Certain QoS features can only be used in either the input or output direction. For more information about these keywords and the QoS features supported, see the "input and output Keywords of the service-policy Command" section on page 5.

Also, if you attach a traffic policy to an interface containing multiple EVCs, the traffic policy will be attached to *all* of the EVCs on the interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-type interface-number*
4. **service instance** *id* **ethernet** [*evc-name*]
5. **encapsulation dot1q** *vlan-id* [,*vlan-id* [-*vlan-id*]] [**native**]
6. **rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric**
7. **bridge domain** *bridge-number*
8. **service-policy** {**input** | **output**} *policy-map-name*
9. **end**
10. **show policy-map interface** *type number* **service instance** *service-instance-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-type interface-number* **Example:** `Router(config)# interface gigabitethernet 0/0/1`	Configures an interface type and enters interface configuration mode. • Enter the interface type and interface number.

	Command or Action	Purpose
Step 4	**service instance** *id* **ethernet** [*evc-name*] **Example:** `Router(config-if)# service instance 333 ethernet evc1`	Configures an Ethernet service instance on an interface and enters Ethernet service configuration mode. • Enter the service instance identification number and, if applicable, the EVC name (optional).
Step 5	**encapsulation dot1q** *vlan-id* [,*vlan-id*[-*vlan-id*]] [**native**] **Example:** `Router(config-if-srv)# encapsulation dot1q 10`	Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance.
Step 6	**rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric** **Example:** `Router(config-if-srv)# rewrite ingress tag translate 1-to-1 dot1q 300 symmetric`	Specifies the encapsulation adjustment to be performed on a frame ingressing a service instance.
Step 7	**bridge domain** *domain-number* **Example:** `Router(config-if-srv)# bridge domain 1`	Configures a bridge domain. • Enter the bridge domain number.
Step 8	**service-policy** {**input** | **output**} *policy-map-name* **Example:** `Router(config-if-srv)# service-policy input policy1`	Attaches a policy map to an interface. • Enter either the **input** or **output** keyword and the policy map name.
Step 9	**end** **Example:** `Router(config-if-srv)# end`	(Optional) Returns to privileged EXEC mode.
Step 10	**show policy-map interface** *type number* **service instance** *service-instance-number* **Example:** `Router# show policy-map interface gigabitethernet 1/0/0 service instance 30`	(Optional) Displays the statistics and the configurations of the input and output policies that are attached to an interface. • Enter the interface type, interface number, and service instance number.

Configuration Examples for EVC Quality of Service

Example: Creating a Traffic Class for Use on the EVC

In this example, traffic with a CoS value of 2 is placed in the traffic class called class1:

```
Router> enable
Router# configure terminal
Router(config)# class-map match-any class1
Router(config-cmap)# match cos 2
Router(config-cmap)# end
```

Example: Creating a Traffic Policy (Policy Map) for Use on the EVC

In this example, traffic policing has been configured in the policy map called policy1. Traffic policing is the QoS feature applied to the traffic in class1:

```
Router> enable
Router# configure terminal
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# police 3000
Router(config-pmap-c)# end
```

Example: Configuring the EVC and Attaching a Traffic Policy to the EVC

In this example, an EVC has been configured and a traffic policy called policy1 has been attached to the EVC:

```
Router> enable
Router# configure terminal
Router(config)# interface gigabitethernet 0/0/1
Router(config-if)# service instance 333 ethernet evc1
Router(config-if-srv)# encapsulation dot1q 10
Router(config-if-srv)# rewrite ingress tag translate 1-to-1 dot1q 300 symmetric
Router(config-if-srv)# bridge domain 1
Router(config-if-srv)# service-policy input policy1
Router(config-if-srv)# end
```

Example: Verifying the Traffic Class and Traffic Policy Information for the EVC

The following is sample output of the **show policy-map interface service instance** command. It displays the QoS features configured for and attached to the EFP on the GigabitEthernet interface 1/1/7.

```
Router# show policy-map interface gigabitethernet 1/1/7 service instance 10

 GigabitEthernet1/1/7: EFP 10

  Service-policy input: multiaction

    Class-map: c1 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 3
      police:
          cir 300000 bps, bc 2000 bytes
        conformed 0 packets, 0 bytes; actions:
          set-prec-transmit 7
          set-qos-transmit 10
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0000 bps, exceed 0000 bps

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
Packet classification	"Classifying Network Traffic" module
Packet marking	"Marking Network Traffic" module
MQC	"Applying QoS Features Using the MQC" module.
EVC connections	"Configuring Ethernet Virtual Connections on the Cisco ASR 1000 Router" module

Standards

Standards	Title
MEF 6.1	*Metro Ethernet Services Definitions Phase 2 (PDF 6/08)*

MIBs

MIBs	MIBs Link
• CISCO-EVC-MIB • CISCO-CLASS-BASED-QOS-MIB	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring EVC Quality of Service

Table 3 lists the release history for this feature.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 3 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 3 Feature Information for EVC Quality of Service

Feature Name	Releases	Feature Information
EVC Quality of Service	Cisco IOS XE Release 3.3	This document contains information about how to enable quality of service (QoS) features (such as traffic classification and traffic policing) for use on an Ethernet virtual circuit (EVC). The EVC Quality of Service feature was introduced on the Cisco ASR 1000 Series Aggregation Services Router. The following commands were introduced or modified: **service-policy**, **show policy-map interface service instance**.