Configuring QoS

This chapter provides information about configuring the Quality of Service (QoS) features on the Cisco 800M Series ISR and contains the following sections:

Configuring Class Based Weighted Fair Queuing

Class Based Weighted Fair Queuing (CBWFQ) provides congestion-management support for user-defined traffic classes. For CBWFQ, you define traffic classes based on match criteria including protocols, access control lists (ACLs), and input interfaces. Packets satisfying the match criteria for a class constitute the traffic for that class. A FIFO queue is reserved for each class, and traffic belonging to a class is directed to the queue for that class.

Once a class has been defined according to its match criteria, you can assign it characteristics. To characterize a class, you assign it bandwidth, weight, and maximum packet limit. The bandwidth assigned to a class is the guaranteed bandwidth delivered to the class during congestion.

For more information about configuring CBWFQ see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_conmgt/configuration/15-mt/qos-conmgt-15-mt-book/qos-conmgt-cfg-wfq.html

Example: Class Based Weighted Fair Queuing

In this example, two class maps are created and their match criteria are defined. For the first class map called class1, the numbered ACL 101 is used as the match criterion. For the second map class, called class2, the numbered ACL 102 is used as the match criterion. Packets are checked against the contents of these ACLs to determine if they belong to the class.

```
Router# configure terminal
Router(config)# access-list 101 permit udp host 10.10.10.10 host 10.10.10.20 range 16384
20000
Router(config# access-list 102 permit udp host 10.10.10.10 host 10.10.10.20 range 53000
56000
Router(config)# class-map class1
Router(config-cmap)# match access-group 101
Router(config-cmap)# exit
Router(config-cmap)# class-map class2
Router(config-cmap)# match access-group 102
Router(config-cmap)# exit
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth 3000
Router(config-pmap-c)# queue-limit 30
Router(config-pmap-c)# exit
Router(config-pmap)# class class2
Router(config-pmap-c)# bandwidth 2000
Router(config)# interface gigabitethernet 0/4
Router(config-if)# service output policy1
Router(config-if)# exit
```

Configuring Low-Latency Queueing

Strict priority queueing allows delay-sensitive data such as voice to be dequeued and sent before packets in other queues are dequeued. Low Latency Queuing (LLQ) provides strict priority queueing for CBWFQ, reducing jitter in voice conversations. LLQ enables use of a single, strict priority queue within CBWFQ at the class level, allowing you to direct traffic belonging to a class to the CBWFQ strict priority queue. To enqueue class traffic to the strict priority queue, you specify the named class within a policy map and then configure the priority command for the class. Within a policy map, you can give priority status to one or more classes. When multiple classes within a single policy map are configured as priority classes, all traffic from these classes is enqueued to the same, single, strict priority queue.

For more information on configuring low latency queuing see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_conmgt/configuration/15-mt/qos-conmgt-15-mt-book/qos-conmgt-cfg-wfq.html

Example: Low-Latency Queueing

```
Router# configure terminal
Router(config)# class-map voice
Router(config-cmap)# match access-group 102
Router(config)# policy-map policy1
Router(config-pmap)# class voice
Router(config-pmap-c)# priority 50 60
Router(config-pmap)# class bar
Router(config-pmap-c)# bandwidth 20
Router(config-pmap)# class class-default
```

```
Router(config-pmap-c)# fair-queue
Router(config)# interface serial 0/0/0
Router(config-if)# service-policy output policy1
Router(config-if)# exit
```

Configuring Class-Based Traffic Shaping

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

For more information on class-based traffic shaping, see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfcbshp.html

Example: Class-Based Traffic Shaping

The following example defines a class c1 which is configured to shape traffic to 384 kbps, with a normal burst size of 15440 bits.

```
Router# configure terminal
```

```
Router(config)# policy-map shape
Router(config-pmap)# class c1
Router(config-pmap-c)# shape average 384000 15440
Router(config-pmap-c)# end
Router(config)# interface Serial 0/0/0
Router(config-if)# service out shape
```

Configuring Class-Based Traffic Policing

Class-based trafic policing allows you to control the maximum rate of traffic transmitted or received on an interface. Class-based traffic policing is often configured on interfaces at the edge of a network to limit traffic into or out of the network. In most class-based policing configurations, traffic that falls within the rate parameters is transmitted, whereas traffic that exceeds the parameters is dropped or transmitted with a different priority.

For more information on configuring class-based traffic policing, see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_plcshp/configuration/15-mt/qos-plcshp-15-mt-book/qos-plcshp-class-plc.html

Example: Class-Based Traffic Policing

In this example, Class-Based Policing is configured with the average rate at 8000 bits per second and the normal burst size at 1000 bytes for all packets leaving Gigabit Ethernet interface 0/4.

```
Router# configure terminal
Router(config)# class-map access-match
Router(config-cmap)# match access-group 1
Router(config)# policy-map police-setting
Router(config-pmap)# class access-match
```

```
Router(config-pmap-c)# police 8000 1000 1000 conform-action transmit exceed-action drop
Router(config-pmap-c)# violate-action drop
Router(config)# interface gigabitethernet 0/4
Router(config-if)# service-policy output policy-setting
Router(config-if)# exit
```

Configuring Class-Based Weighted Random Early Detection

Weighted Random Early Detection (WRED) combines the capabilities of the Random Early Detection (RED), algorithm with the IP Precedence feature to provide for preferential traffic handling of higher priority packets. WRED can selectively discard lower priority traffic when the interface begins to get congested and provide differentiated performance characteristics for different classes of service.

You can configure WRED to ignore IP precedence when making drop decisions so that nonweighted RED behavior is achieved.WRED makes early detection of congestion possible and provides for multiple classes of traffic. It also protects against global synchronization. For these reasons, WRED is useful on any output interface where you expect congestion to occur.

For more information about configuring WRED, see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_conavd/configuration/15-mt/qos-conavd-15-mt-book/qos-conavd-cfg-wred.html

Example: Class-Based Weighted Random Early Detection

```
Router(config-if)# class-map c1
Router(config-cmap)# match access-group 101
Router(config-if)# policy-map p1
Router(config-pmap)# class c1
Router(config-pmap-c)# bandwidth 48
Router(config-pmap-c)# random-detect dscp-based
Router(config-pmap-c)# random-detect dscp 8 24 40
Router(config-if)# service-policy output p1
```

Configuring QoS Hierarchical Queueing Framework

The QoS Hierarchical Queueing Framework (HQF) feature enables you to manage quality of service (QoS) at three different levels: the physical interface level, the logical interface level, and the class level for QoS queueing and shaping mechanisms by using the modular QoS command-line interface (MQC) to provide a granular and flexible overall QoS architecture.

For more information about configuring hierarchical queueing framework see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_hrhqf/configuration/15-mt/qos-hrhqf-15-mt-book/qos-hrhqf.html

Configuring Network-Based Application Recognition

Network-Based Application Recognition (NBAR) is a classification engine that recognizes and classifies a wide variety of protocols and applications. When NBAR recognizes and classifies a protocol or application, the network can be configured to apply the appropriate quality of service (QoS) for that

application or traffic with that protocol.

For more information about configuring NBAR, see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/configuration/15-mt/qos-nbar-15-mt-book.html

Example: Network Based Application Recognition

```
Router# configure terminal
Router(config)# class-map cmap1
Router(config-cmap)# match protocol citrix
Router(config-cmap)# end
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Device(config-pmap-c)# bandwidth percent 50
Device(config-pmap-c)# end
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 0/4
Device(config-if)# service-policy input policy1
Device(config-if)# end
```

Configuring Resource Reservation Protocol

Resource Reservation Protocol (RSVP) is the industry-standard protocol for dynamically setting up end-to-end QoS across a heterogeneous network. RSVP, which runs over IP, allows an application to dynamically reserve network bandwidth. Using RSVP, applications can request a certain level of QoS for a data flow across a network.

The Cisco IOS QoS implementation allows RSVP to be initiated within the network using configured proxy RSVP. Using this capability, you can take advantage of the benefits of RSVP in the network even for non-RSVP enabled applications and hosts. RSVP is designed to guarantee network bandwidth from end-to-end for IP networks.

For more information about configuring RSVP, see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_rsvp/configuration/15-mt/qos-rsvp-15-mt-book/config-rsvp.html

Configuring Quality of Service for VPNs

The QoS for VPNs feature provides a solution for making Cisco IOS QoS services operate in conjunction with tunneling and encryption on an interface. Cisco IOS software can classify packets and apply the appropriate QoS service before the data is encrypted and tunneled. The QoS for VPN feature allows users to look inside the packet so that packet classification can be done based on original port numbers and based on source and destination IP addresses. This allows the service provider to treat mission critical or multi-service traffic with higher priority across their network.

For more information about configuring QoS for VPNs see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_classn/configuration/15-mt/qos-classn-15-mt-book/qos-classn-vpn.html

Configuring Per Tunnel QoS for DMVPN

The Per-Tunnel QoS for DMVPN feature lets you apply a quality of service (QoS) policy on a Dynamic Multipoint VPN (DMVPN) hub on a per-tunnel instance (per-spoke basis) in the egress direction for DMVPN hub-to-spoke tunnels. The QoS policy on a DMVPN hub on a per-tunnel instance lets you shape tunnel traffic to individual spokes (a parent policy) and differentiate individual data flows going through the tunnel for policing (a child policy). The QoS policy that the hub uses for a specific spoke is selected according to the specific Next Hop Resolution Protocol (NHRP) group into which that spoke is configured. Although you can configure many spokes into the same NHRP group, the tunnel traffic for each spoke is measured individually for shaping and policing.You can use this feature with DMVPN with or without Internet Protocol Security (IPsec).

For more information about configuring Per-Tunnel QoS for DMVPN, see the following web link:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_conn_dmvpn/configuration/15-mt/sec-conn-dmvpn-15-mt-book/sec-conn-dmvpn-per-tunnel-qos.html

Configuring Layer 2 Auto QoS

You can use the auto-QoS feature to simplify the deployment of QoS features. Auto-QoS determines the network design and enables QoS configurations so that the router can prioritize different traffic flows. It uses the ingress and egress queues instead of using the default (disabled) QoS behavior. The switch offers best-effort service to each packet, regardless of the packet contents or size, and sends it from a single queue.

When you enable auto-QoS, it automatically classifies traffic based on the traffic type and ingress packet label. The switch uses the classification results to choose the appropriate egress queue.

For more information about configuring Auto QoS, see the following link:

http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3750/software/release/12-2_55_se/configuration/guide/scg3750/swqos.html#wp1231112