Priority Queuing

Priority queuing is a platform-independent congestion management method that, as its name implies, provides some packets absolute priority over other packets. The simplest analogy is a grocery store with two checkout lines being serviced by one clerk.

Customers holding a special customer club card can enter one line, and the remaining customers can enter the second line. The clerk cannot check out any customers in the second line until he serves all the preferred customers (the ones holding the club cards in the first line).

This is exactly how routers implement priority queuing. Packets are classified based on a configurable policy and are placed in a queue. The IOS implementation has a total of four queues: high, medium, normal, and low.

The high queue must be empty before the medium queue can be serviced, the medium queue must be empty before the normal queue can be serviced, and the normal queue must be empty before the low queue can be serviced.

This method has one drawback—and it's easy to see if we go back to the grocery store example for a moment. If you're forced to enter the non–club card line, you could wait quite some time before the clerk finally gets around to checking you out. In fact, you might *never* get checked out if enough people continue to arrive in the club card line. You could starve standing there!

It's no different with priority queuing. Lower priority queues actually can starve—never being allowed to send any traffic. Because of this, priority queuing should be implemented with care. A packet classification policy should be chosen so it doesn't place packets in the high priority queue at a very high rate.

Configuring and Monitoring Priority Queuing

Example 8-1 demonstrates how to configure priority queuing.

Example 8-1. Configuring Priority Queuing

access-list 110 permit ip host 10.1.1.1 any access-list 120 permit ip host 172.16.28.1 any access-list 130 permit ip any any ! priority-list 1 protocol ip high list 110 priority-list 1 protocol ip medium list 120 priority-list 1 protocol ip normal list 130 ! interface Serial0 priority-group 1

For example, say a router has the following packets to transmit on Serial 0:

Packet 1—

Sourced from 172.18.40.8, destined to 192.168.40.8

Packet 2—

Sourced from 10.1.1.1, destined to 192.168.40.8

Packet 3—

Sourced from 172.18.40.8, destined to 192.168.10.1

Packet 4—

Sourced from 10.1.1.1, destined to 192.168.10.1

Packet 5—

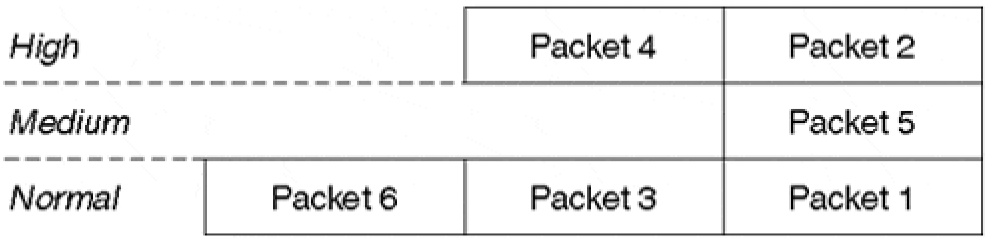
Sourced from 172.16.28.1, destined to 192.168.10.1

Packet 6—

Sourced from 172.18.40.8, destined to 192.168.20.56

These packets first are enqueued into the high, medium, and normal queues based on the access lists referenced in the **priority-list** commands, with the results as shown in Figure 8-2.

Figure 8-2. Packets Enqueued Based on the *priority-list* Command



After the packets are placed into the correct queues, IOS services the high queue until no packets remain—packets 2 and 4 in this case. IOS then transmits the packets in the medium queue—packet 5 here. Finally, when the high and medium queues are empty, the packets remaining in the normal queue are transmitted.

Of course, while these packets are being transmitted, other packets continue to arrive and are placed into their appropriate queues. If enough packets enter the high and medium queues that the IOS never actually transmits anything out of the normal queue, the normal queue starves. Any applications or hosts that use the normal queue to pass traffic simply fail to function.

You can examine the priority queuing configuration using the **show queueing priority** command, as demonstrated in Example 8-2.

Example 8-2. *show queuing priority* Displays Priority Queuing Status

router#**show queueing priority** Current priority queue configuration: List Queue Args 1 high protocol ip list 110 1 medium protocol ip list 120 1 normal protocol ip list 130

The output of the **show queueing priority** command summarizes the current priority queuing configuration. It doesn't provide any information about the number of packets that were placed into or dropped from any given queue. To determine the number of packets that were passed through or dropped from each queue, examine the output of the **show interface** command as demonstrated in Example 8-3.

Example 8-3. *show interface* Displays Packet Traffic through a Queue

router#**show interface serial 0** Serial0 is up, line protocol is up …. Queueing strategy: priority-list 1 Output queue (queue priority: size/max/drops): high: 0/20/0, medium: 0/40/0, normal: 0/60/0, low: 0/80/0

From the output in Example 8-3, you can see the type of queuing enabled (in the **Queueing strategy** area), the specific priority group that is assigned to this interface, the number of packets currently in each of the queues, and the number of packets dropped from each queue.

The maximum size of each queue can be configured using the IOS **priority-list***list-number* **queue-limit** configuration command.

Last updated on 12/5/2001
Inside Cisco IOS Software Architecture, © 2002 Cisco Press

Index terms contained in this section