

Queuing and Compression

Assists in administrating congested WAN bandwidth

Cisco IOS Queuing Options

Queuing is the management of how data is removed from the buffers and placed on to the wire. When there is more data than bandwidth, the excess data is stored in the buffer. Without queuing the data is removed from the buffer in the order in which it was placed into the buffer or better know as First-In First-Out (FIFO).

First-in, First-out (FIFO) - Transmission occurs in the same order as messages are received.

Weighted fair queuing (WFQ) - Session basis. Conversation prioritizes interactive traffic over file transfers to ensure satisfactory response time for common user applications. WFQ is **on by default** for all interfaces having a line equal to or lower than 2.048Mbps (E-1 speed).

Priority queuing - Packet basis. Ensures timely delivery of a specific protocol or type of traffic, port, that is transmitted before all others. Uses four different priorities: high, medium, normal, and low.

Custom queuing - Establishes **bandwidth allocations** for each different type of traffic. Uses virtual pipes. Permits a maximum of 16 configurable queues.

Weighted Fair Queuing Implementation

```
>int s0
>fair-queue [ congestive discard threshold ] { dynamic-queues } { reservable-queues }
(Discard threshold value can be 1 to 512, default is 64. Specifies the number of
conversations that can exist within the queue. If exceeded, those conversations won't be
allocated their equal amount of bandwidth.)
>sh queueing
>sh queue (interface)
```

Priority Queuing Implementation

1. Create a priority list for the packet priority.
2. Adjust the size of the queues if desired.
3. Apply the priority list to an interface.

```
>priority-list list-number [protocol protocol-name] [interface interface-name]
(high | medium | normal | low) default | queue-limit
```

list number = 1-16

```
>priority-list 1 protocol ip low gt 1400
```

```
>priority-list 1 protocol ip high
```

```
>priority-list 1 interface serial 1 normal
```

```
>priority-list 1 interface ethernet0/3 high
```

```
>priority-list 1 queue-limit 20 30 60 120
```

```
>int s 0
```

```
>priority-group 1 (implicitly created)
```

```
>sh queueing priority
```

Serial Data Compression

TCP Header Compression - Cisco uses the Van Jacobson algorithm for compression of the IP packets. Doesn't compress MAC, layer-2, addresses.

```
>int s0
```

```
>ip tcp header-compression <passive>
```

passive - only outbound TCP traffic if the data received is compressed

Payload Compression - "per-virtual" circuit. Compresses the payload of the header. Used for point-to-point circuits.

Link Compression - Compresses the payload of the header.

compression algorithms: Stacker, Predictor, and LZS.

Stacker - Compresses PPP or LAPB. Cisco HDLC.

Predictor - Compresses PPP or LAPB. Cisco HDLC.

LZS - Compresses PPP or LAPB. Cisco HDLC.

and "predicts" the next byte.

replaces the next byte with the predicted byte.

```
>int s0
```

```
>compress lzss
```