Quality of Service

Jitter: the variation in the latency of the received packets

Capacity: How may bits (packed into packets) can be received per second

Latency cannot be solved above Layer 4, it has to be solved at layer 3 and below.

Jitter is a problem for real time applications that are expecting to receive a packet in a certain time window. Can be solved with buffers but the consequence is an increase to latency.

Packet Loss is catastrophic for files

Packet loss is not too bad for voice or video. Better to loose a frame than receive a packet after it's of any use. These formats do not use TCP.

Capacity is a layer one issue but it is limited by the slowest link and so layers 2 and 3 come into play.

The solution to many of our problems today is to increase the capacity. Then there is little or no queuing at routers/switches, so no congestion.

It is prohibitively expensive to ensure that there is a over supply of capacity at all links at all times.

Statistical multiplexing of packets. Not all applications have the same requirements and so prioritization can take place.

In times of congestion where statistical multiplexing is happening, we give priority to those packets that will suffer the most from increased packet loss, latency and jitter.

The priority section of a packet header at each layer can be used to determine the quality of service metric. Do not do this at the application layer (everyone just sets the packet to the highest priority).

User applications are un-trusted. The entity purchasing the service may get to decide the priority but it will then be checked by the Network Provider for the service.

QoS tool

Can be any number of tools

- Policer: discarding packets above a threshold
- Shaper: Delay packets so as to not go above the bandwidth threshold
- Classifier: Inspects an incoming packet and assigns it to a class of service
- Metering and Coloring: Check the rate of packet arrival against thresholds and then assign them to classes/colors
- Queuing differentiation: Queues all work in FIFO mode, but packets with different class of service can be assigned different queues.

- Scheduler: decides in which order to get packets from the queues.
- Rewrite: modify a packets COS marking.

Classifier

In an ideal world a router would receive a packet, inspect it, and assign its COS marking to the routers COS class. But in practice three actions can happen.

- Trust and use for internal operations.
- Increase the granularity (Give more depth to the packet differentiation)
- Change the COS marking completely

Metering and Coloring

Customer SLAs with service provider. (Bandwidth and so on)

- The customer will have a committed information rate
- They might also have a peak information rate (if capacity is available it will be satisfied)
- Any packet above the threshold will be dropped.

Two rate colour marking

Colour Blind All input traffic is the same

- Data below CIR marked green
- Data above CIR marked yellow
- Above PIR marked red (dropped)

Green goes through CIR port and Yellow through PIR port

Colour Aware

Input already marked (green, yellow)

- green goes into the CIR queue, but if above CIR rate it is sent to the PIR queue
- Yellow goes to PIR queue
- Traffic above PIR dropped
- Yellow never becomes green even if space in CIR queue.

Policer

Performs actions on packets based on a data rate threshold limit

Traffic above a certain rate is dropped, typically performed at the ingress of the router.

Traffic sources that are aware of the contract that is being policed may perform traffic shaping to ensure that the traffic always complies to the contract.

Shaper

Performs actions on packets based on the data rate threshold.

Action is to delay the packet instead of dropping. Ensuring the threshold is never exceeded.

Policer implementation: Token Bucket

Tokens are periodically added to the system and each incoming packet has two options.

1. If credit available: Enter one of the queues (credit consumed)
2. If no credits: Discarded (No credit consumed)

Bucket depth indicates the maximum number of credits that can be stored at any given time.

Number of tokens per time determined the bandwidth limit.

Shaper implementation: leaky bucket

Can store packets in a buffer at whatever rate they are arriving at.

The buffer depletes at a constant rate, determined by the bandwidth limit.

There does have to be a size to the buffer and so in reality there will be some point at which packets start to be dropped.

This will increase latency!

Queues

FIFO but some actions can be taken to drop packets before the queue is 100% full

Each router typically has multiple ports which can be used to send data out of.

For example MPLS has 3 bits meaning 8 priority differentiating opportunities that can be mapped to ports, whereas DSCP has 6 bits meaning 64 possibilities.

Scheduler

Used with the differentiated queues, implementing an algorithm to decide how to best serve the various queues

For example first serving the packets in the queue with the highest priority.

Rewrite

Changing the COS marking of an incoming packet.

Can be used due to trust issues. The system may not trust the source of the packet.

Can also be used to signal information to the downstream router, e.g to packets arrive marked 1 but one is changed to 2 to indicate that the packet has exceeded the agreed rate i.e green and yellow.

Effects of QoS on packet loss, jitter and latency

Packet loss caused by:

- Traffic policer, drop by the queue

Latency caused by:

- Shaper, scheduler

Jitter:

- Shaper, scheduler

Queues Types

1. FIFO
2. Fair Queuing
3. Strict Priority
4. Weighted Fair Queuing
5. Weighted Round Robin
6. Deficit Weighted Round Robin
7. Priority-Based Deficit Weighted Round Robin

Fair Queueing (FQ)

Separates the different floes into separate logical flows

Scheduler picks a packet from a logical queue in a round robin order.

No quality of service differentiation as it is fair to all flows.

Not really used as flow separation is too expensive.

Strict priority (SP) and WFQ

SP: Packets are taken from the queue of the highest priority then second and so on.

Low priority traffic can become stalled.

WFQ: assigns weights proportional to the priority. Not waiting until the queue is empty

Operates bit by bit to avoid packet size influence on the transission rate from a queue

Weighted round robbin

Same as WFQ but considers packets instead of bits

Not as fair but less computationally expensive.

Deficit Weight round Robbin

Assigns tokens to queues proportional to their weight. when a packet is transmitted then is consumes tokens proportional to the size f the packet.

Queue is skipped when not enough quantums to transmit packet.

PB-DWRR

Adds a stricter weighting to packets that ned les latency and jitter.

Some qeues are served strictly before others and some follow DWRR

Can be an issue if the high priority monopolises the connection. Police the priority queue.

Early dropping

1. Random Early Discard
2. Weighted Random Early Discard

RED

Drop packets at some probabilistic ratio before the queue fills up. Sending TCP a message (it will lower the transmission rate). This is better than all of a sudden dropping all the packets.

Also Weighted Random Early Discard WRED. giving lower priority queues a higher drop percentage

Acronyms

QoS: Quality of Service COS: Class of Service CIR: Committed information Rate PIR: Permitted Information Rate FQ: Fair Queue SP: Strict Priority WRR: Weighted round robbin DWRR: Deficient Weighted Round robbin PB-DWRR: Priority Based Deficit Weighted Round Robbin RED: Random Early Discard WRED: Weighted RED DSCP: Differentiated Services Code Point