Differentiated Services Architectures

Carlo Vallati
PostDoc Researcher@ University of Pisa
c.vallati@iet.unipi.it

Intro

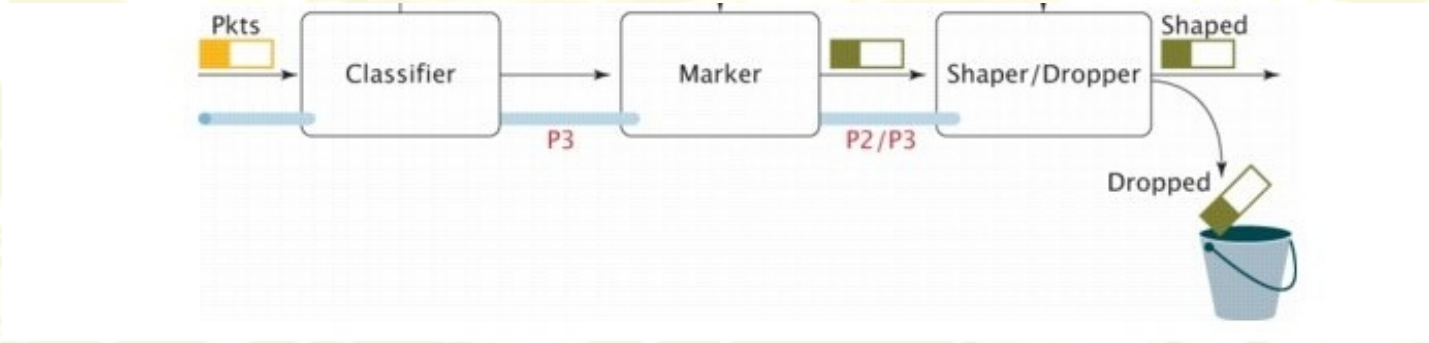
- Network Boundary
  - Classification & Marking
  - Shaping and Policing
- Per-Hop Behavior
  - Scheduling and Resource Allocation
  - Congestion Avoidance and Packet Drop Policy

Network Boundary

Carlo Vallati
PostDoc Researcher@ University of Pisa
c.vallati@iet.unipi.it

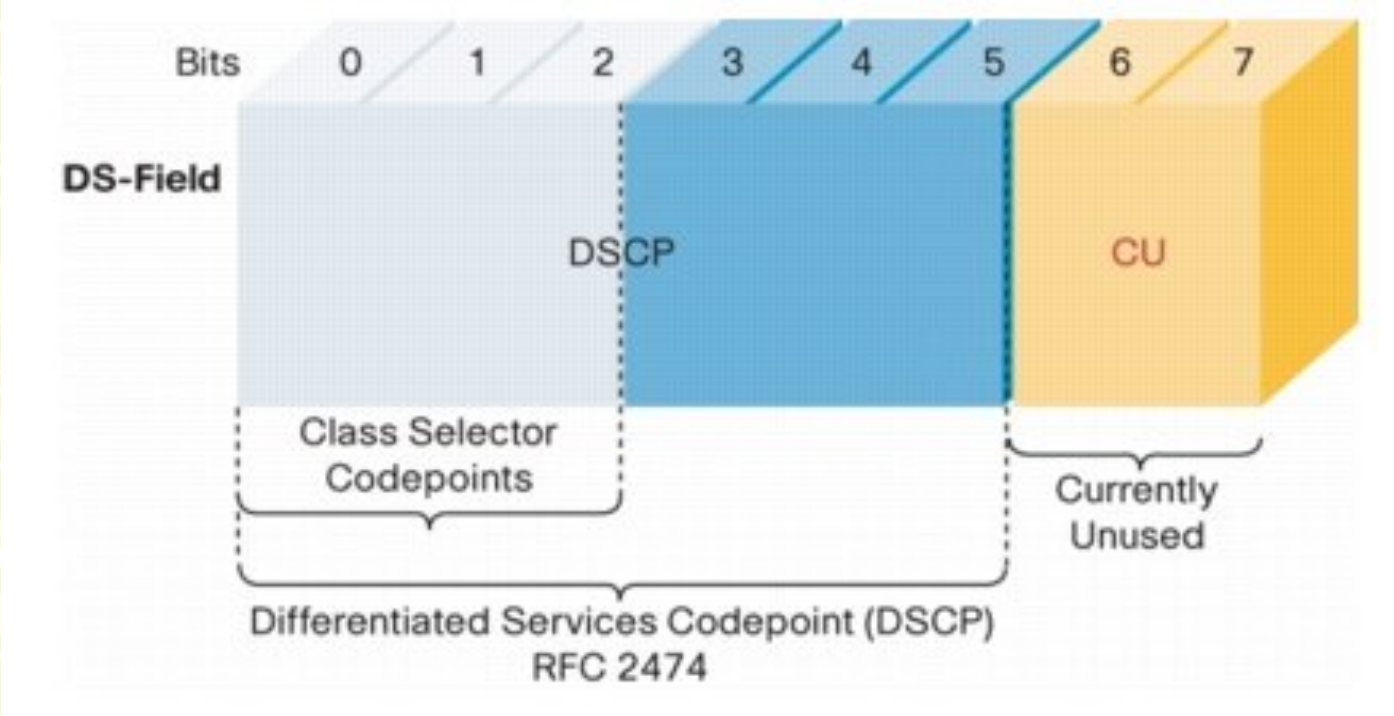
Network Ingress

- Routers at the network boundary perform ***Classifier*** functions to identify packets belonging to a certain traffic class based on one or more TCP/IP header fields
- A ***Marker*** is used to color the classified traffic
- A ***Shaper*** or ***Traffic Policing*** is used to regulate ingress rate



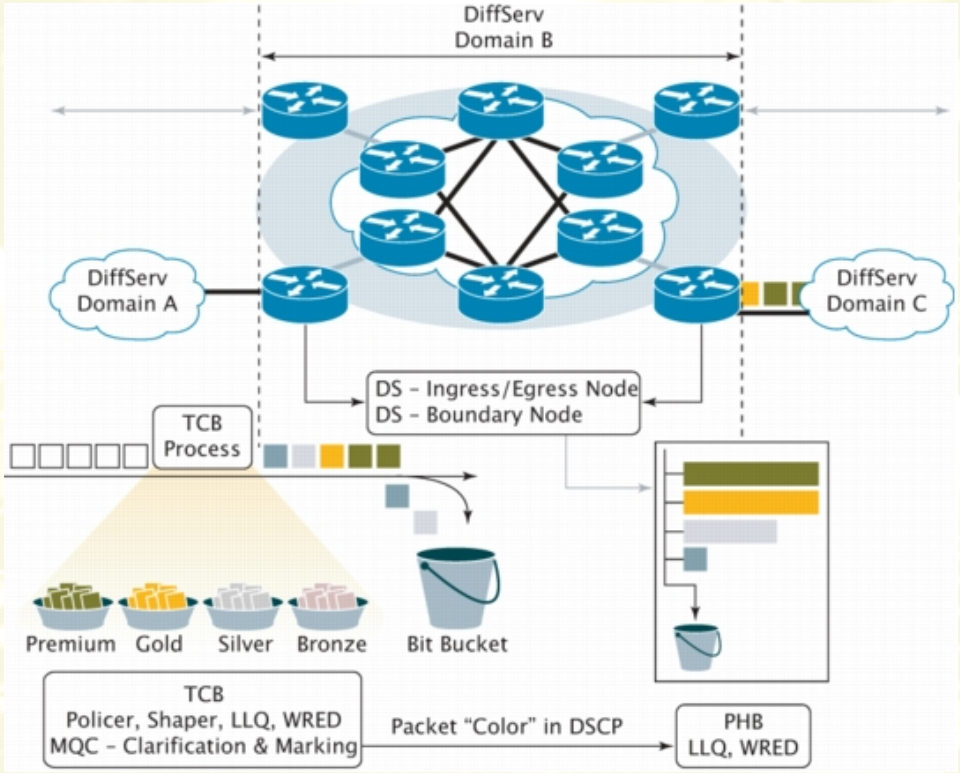
Packet Marker

- The marker set the Differentiated Services Code Point (DSCP) field



Per-Hop Behavior

- Within the network core, a per-hop behavior (PHB) is applied to the packets based on either the IP Precedence or the DSCP field marked in the packet header



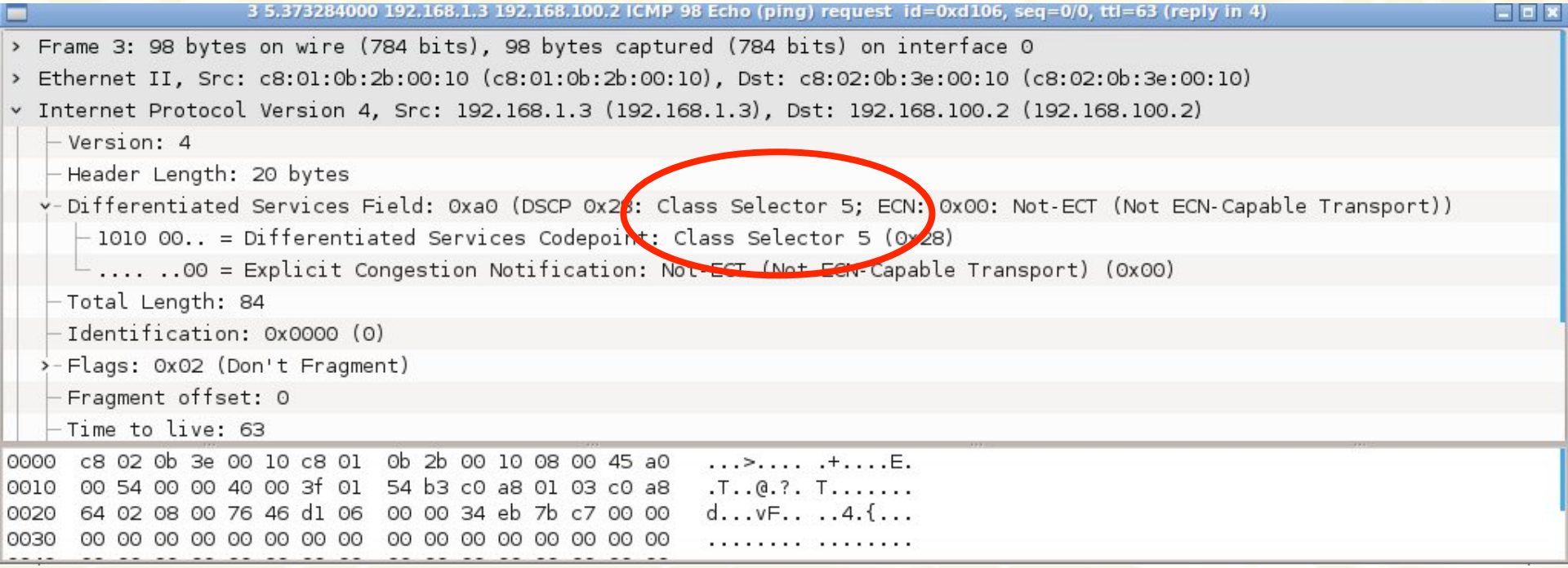
Packet C&M using PBR (Policy Based Routing)

- Define Classification
  - `access-list 1 permit 192.168.1.2 0.0.0.0`
- Define Marking:
  - `interface Ethernet 0/0`
  - `ip policy route-map prio`
  - `route-map prio permit 10`
  - `match ip address 1`
  - `set ip precedence 5`
  - `route-map prio permit 40`
  - `set ip precedence 4`
- To disable
  - `no ip policy route-map prio`

PBR can only set ip priority and ToS

Test

- Test with ping and check that the precedence bit on the IP header is set correctly using wireshark!



Classification based on protocol

- Classify traffic based on transport protocol or application:
  - `access-list 101 permit tcp any any`
  - `access-list 102 permit udp any any`
- Define Marking:
  - `interface Ethernet 0/0`
  - `ip policy route-map prio`
  - `route-map prio permit 20`
  - `match ip address 101`
  - `set ip precedence 6`
  - `route-map prio permit 30`
  - `match ip address 102`
  - `set ip precedence 3`

Test with iperf

iperf generates traffic from client to server!

- Iperf is a tool to generate traffic
- Generate TCP flow (data is generated from client to server)
  - `iperf3 -s (to create the server)`
  - `iperf3 -c 192.168.101.2 (to start the flow)`
- Generate UDP flow (data is generated from client to server)
  - `iperf3 -s (to create the server)`
  - `iperf3 -u -c 192.168.100.2 -b 10M (to start 10Mbps flow)`

Test!

- Start two flows, one UDP and one TCP and check with wireshark the precedence value

Policing/Shaping

Table 3-3. Comparison Between Policing and Shaping Functions	
Policing Function (CAR)	**Shaping Function (TS)**
Sends conforming traffic up to the line rate and allows bursts.	Smoothes traffic and sends it out at a constant rate.
When tokens are exhausted, it can drop packets.	When tokens are exhausted, it buffers packets and sends them out later, when tokens are available.
Works for both input and output traffic.	Implemented for output traffic only.
Transmission Control Protocol (TCP) detects the line at line speed but adapts to the configured rate when a packet drop occurs by lowering its window size.	TCP can detect that it has a lower speed line and adapt its retransmission timer accordingly. This results in less scope of retransmissions and is TCP-friendly.

To limit ingress traffic rate

Smooth traffic flow on an interface to avoid link congestion

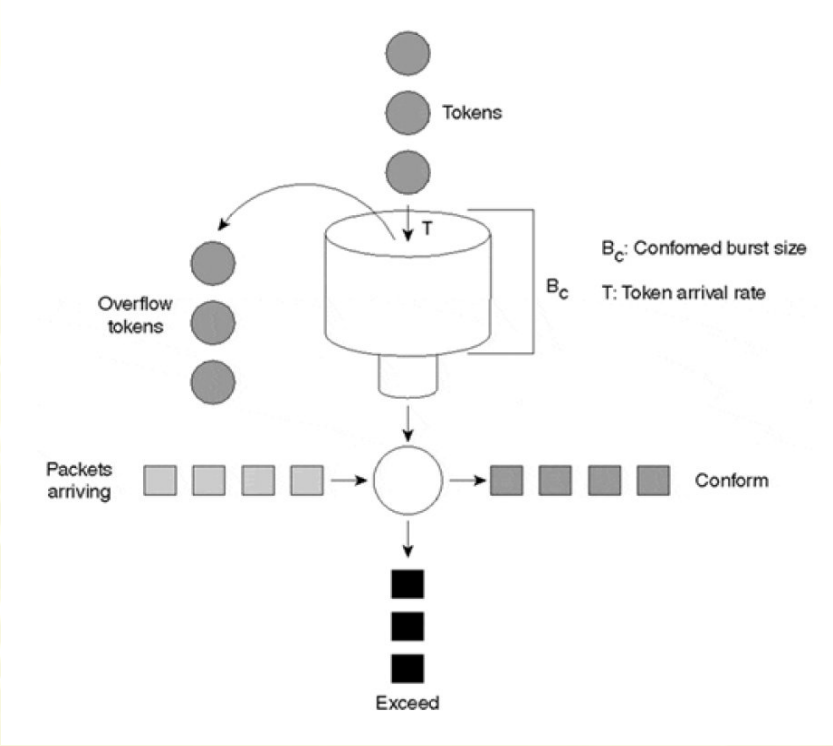
CAR (Committed Access Rate)

- CAR is a traffic Classifier/Marker/Policing
- Usually adopted at the ingress router to limit ingress traffic of a flow
- It performs Classification & Marking & Traffic Policing at the same time

CAR (Committed Access Rate)

- Rate limit statement:
  - `rate-limit {access-group num} <input/output> "CIR" "conformed burst" "extended burst" conformed-action" "action desired" exceed-action "action desired"`

CIR, Committed Information Rate, *bit per second*, average traffic rate
Conformed burst size, *bytes*, amount of traffic allowed to exceed the bucket on an instantaneous basis
Extended burst size, *bytes*, bonus instantaneous rate based on token borrowing mechanism, if set equal to conformed burst size is disabled



Packet C/M/P using CAR

- Define classifier
  - `access-list 101 permit udp 192.168.1.2 0.0.0.0 any`
    - *Policing only udp traffic from 192.168.1.2*
- Define marker and policing function:
  - `interface Ethernet 0/0`
  - `rate-limit input 10000000 2000 2000 conform-action continue exceed-action drop`
    - *Limit all the traffic to 30Mpbs (continue -> check other rules)*
  - `rate-limit input access-group 101 1000000 200000 200000 conform-action set-dscp-transmit cs3 exceed-action drop`
    - *Limit the traffic from 192.168.1.2 to 1Mbps*

Packet C/M/P using CAR

- Show status
  - `show interfaces Ethernet 0/0 rate`

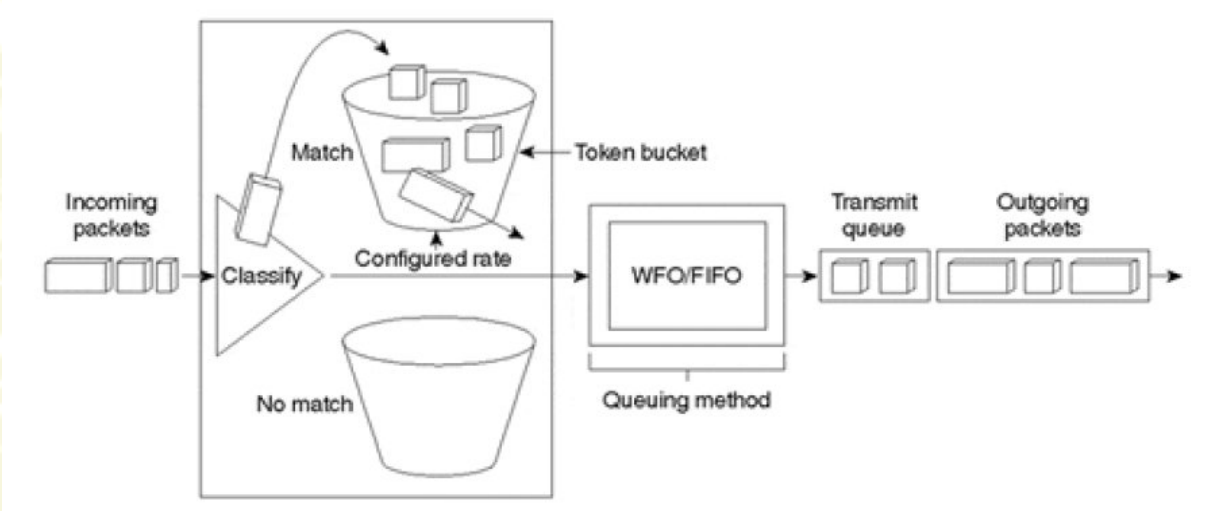
```
Ethernet0/0
  Input
    matches: access-group 101
      params:  1000000 bps, 200000 limit, 200000 extended limit
      conformed 2039 packets, 2885550 bytes; action: set-prec-transmit 2
      exceeded 393 packets, 518162 bytes; action: drop
      last packet: 358299ms ago, current burst: 199897 bytes
      last cleared 00:06:39 ago, conformed 57000 bps, exceeded 10000 bps
R1#
```

Rate Limit Based on Precedence

- Define classifier
  - `access-list 101 permit ip any any precedence 7`
    - *Classify traffic based on its precedence value (or dscp)*
- Define marker and policing function:
  - `interface Ethernet 0/0`
  - `rate-limit input access-group 101 1000000 200000 200000 conform-action transmit exceed-action set-prec-transmit 2`
    - *Change the precedence field when traffic exceed*

Traffic Shaping (TS)

- TS smoothes bursty traffic to meet the configured CIR by queuing or buffering packets exceeding the mean rate (*outbound traffic only*)
- Queued packets are transmitted as tokens become available (*Packets are not discarded*)
- Shaper is usually used at the core to avoid link congestion



Generic Traffic Shaping

- Add shaper
  - `interface Ethernet 1/0`
  - `traffic-shape rate 256000`

  - `show traffic-shape Ethernet 0/0`

- Apply traffic shape selectively
  - `access-list 101 permit ip any any precedence 7`
  - `traffic-shape group 101 rate 256000`

Test!

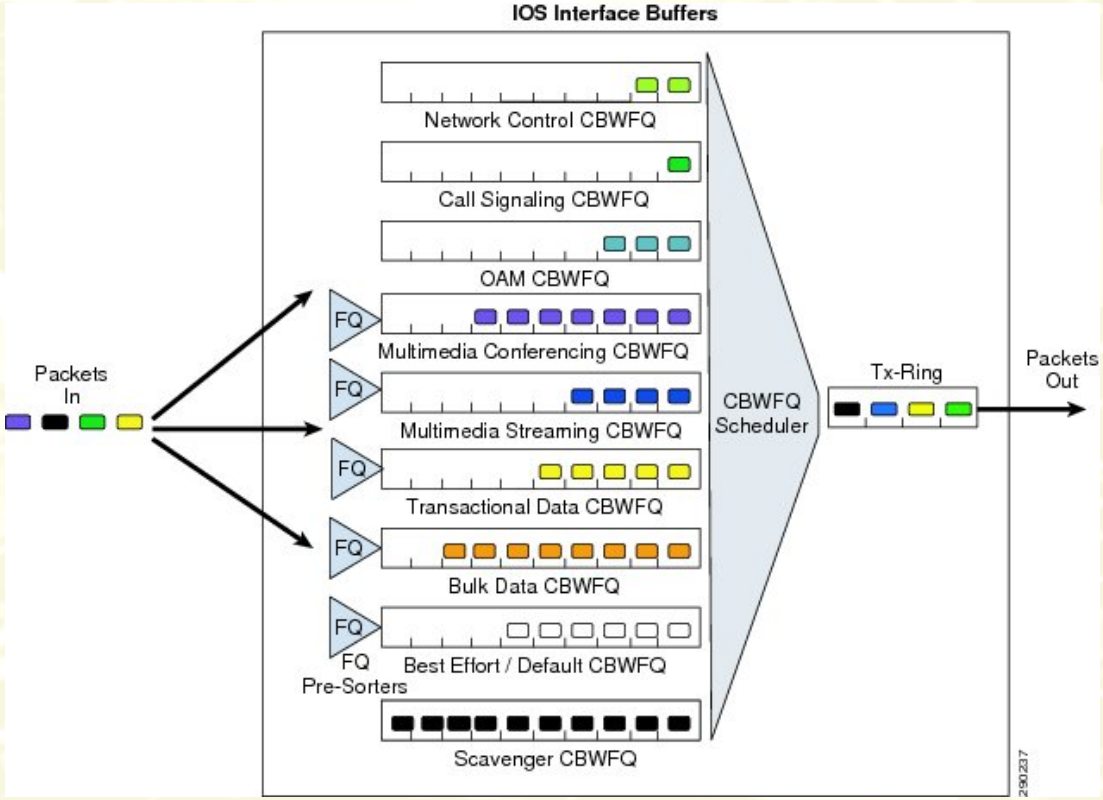
- Add a shaper to the outgoing interface of the ingress node and test it with iperf

Per Hop Behavior

Carlo Vallati
PostDoc Researcher@ University of Pisa
c.vallati@iet.unipi.it

Class-Based WFQ

- Traffic Class are defined
- CBWFQ allocates a different subqueue for each traffic class



CBWFQ

- Class definition
  - access-list 101 permit udp any any range 1500 1600
    - *In the class fit all the UDP traffic using ports in the range 1500 1600 at destination*
  - class-map gold
  - match access-group 101
- Allocate bandwidth
  - policy-map goldservice
  - class gold
  - bandwidth 5000 or bandwidth percent 90
    - Bandwidth in Kbps or in percent of the link capacity
  - interface Ethernet 1/0
  - service-policy output goldservice

CBWFQ - Test

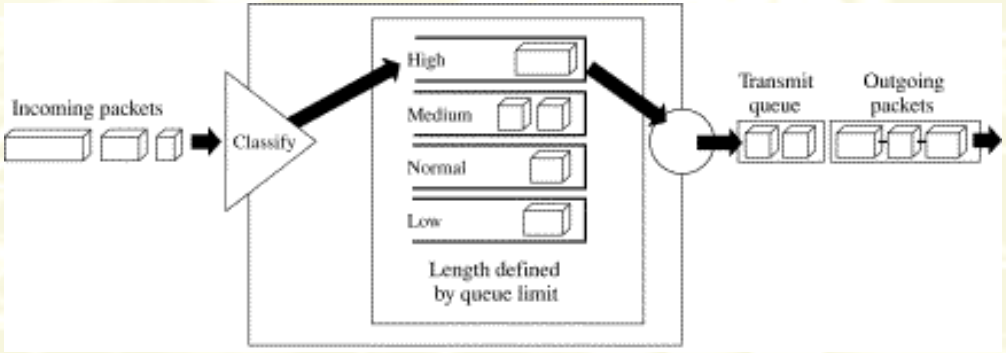
- Run iperf, two servers and two clients
  - `iperf3 -s {-p 1501}`
  - `iperf3 -u {-p 1501} -c 192.168.100.2 -b 10M`
- Get information
  - `show policy goldservice`
  - `show class gold`
- Get statistics
  - `show policy-map interface`

CBWFQ – Class Definition

- Class definition criteria based on ip precedence or dscp field are possible
- Input interface
  - `class-map server`
  - `match input-interface Ethernet 0/0`
  - `policy-map servbandwidth`
  - `class server`
  - `bandwidth 30000`
- Precedence field
  - `class-map class0`
  - `match ip precedence 4`
  - `policy-map class0policy`
  - `class class0`
  - `bandwidth 30000`

Priority Queue

- Priority queuing maintains four output subqueues ***high, medium, normal, and low***
- They are served in decreasing order of priority
- A queue is served only if all higher priority queues are empty
- Low priority queues might ***starve***



Priority Queue

- Classifier
  - `access-list 101 permit ip any any precedence 3`
    - Classify packet based on precedence
  - `access-list 1 permit 192.168.1.2 0.0.0.0`
    - Classify packets based on source address
- Set Priority Queue
  - `priority-list 1 protocol ip high list 101`
  - `priority-list 1 protocol ip medium list 102`
  - `priority-list 1 protocol ip normal list 1`
  - `priority-list 1 protocol ip low list 100`
  - `priority-list 1 default low`
- Set the interface
  - `interface Ethernet 1/0`
  - `priority-group 1`

Priority Queue - Test

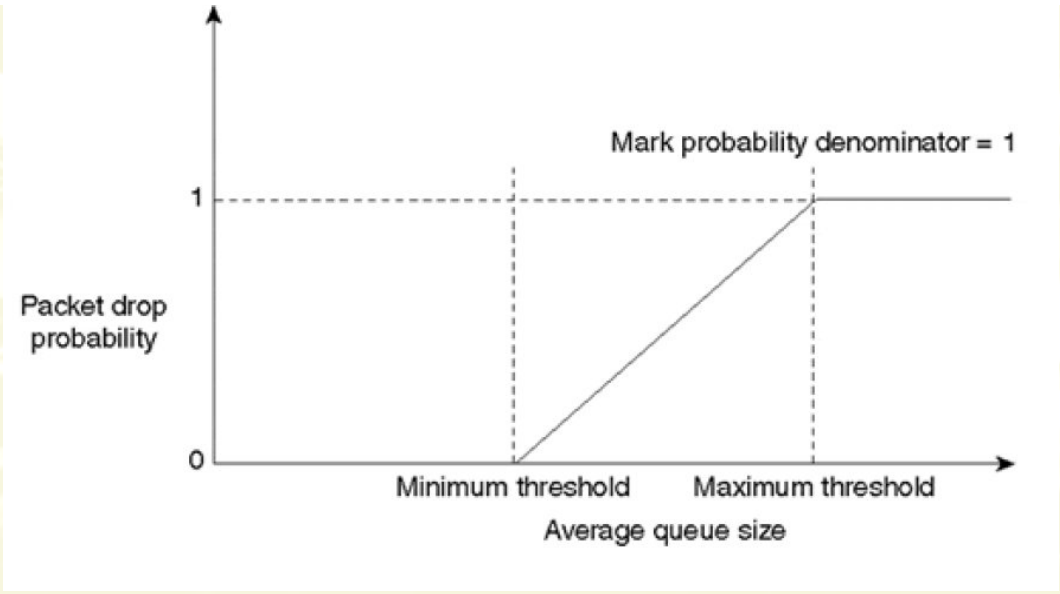
- Start some traffic
  - `iperf -c 192.168.100.2`
  - `iperf -s`

- Priority queue information
  - `show queuing priority`
  - `show interface Ethernet 1/0`

Proactive Queue Management for Congestion Avoidance - RED

- RED is a congestion avoidance mechanism
- RED takes a ***proactive approach***, instead of waiting until the queue is full, it starts dropping packets with ***a non-zero drop probability*** when the average size is above a threshold

$$\text{packet drop probability} = \left(\frac{\text{(average queue length} - \text{minimum threshold)}}{\text{denominator}}\right) \times \text{mark probability}$$



Mark probability denominator is the fraction of packets dropped when the average queue depth is at the maximum threshold. If the mark probability denominator is 10, for example, 1 out of every 10 packets is dropped

Configuring RED

- Enabling RED
  - `interface Ethernet 1/0`
  - `random-detect`
- Tuning RED based on IP precedence
  - `random-detect precedence <prec-value> <minimum-threshold> <maximum-threshold> <mark-prob-den>`
  - `random-detect precedence 5 1000 3000 10`
- Check RED status
  - `show queuing random-detect`

References

- Committed Access Rate: http://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfcar.html
- Generic Traffic Shaping: http://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfgts.html
- DSCP values: http://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/qos-packet-marking/10103-dscpvalues.html
- http://www.cisco.com/en/US/technologies/tk543/tk766/technologies_white_paper09186a00800a3e2f.html
- Policy Based Routing: http://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfpbr.html
- Class Based WFQ: http://www.cisco.com/c/en/us/td/docs/ios/12_0s/feature/guide/fswfq26.html
- Priority Queue: http://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfpq.html
- RED: http://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfwred.html