Network QoS
371-2-0213

Lecture 3

Gabriel Scalosub

Outline

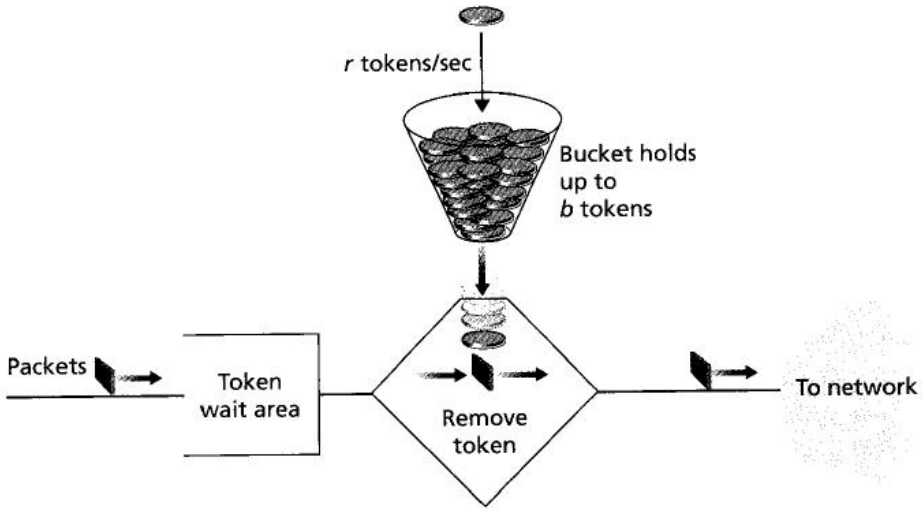
IntServ: Bandwidth vs. Delay

- Main concern: worst-case delay
- Main mechanism: BW reservation
- Protocol: RSVP
  - receiver oriented
  - collect data along path
  - allow receiver to calculate required BW
- E2E queuing delay (with error terms):

$$\frac{(b-M)(p-R)}{R(p-r)} + \frac{M + C_{\text{tot}}}{R} + D_{\text{tot}} \quad p > R \geq r$$

Token/Leaky Bucket

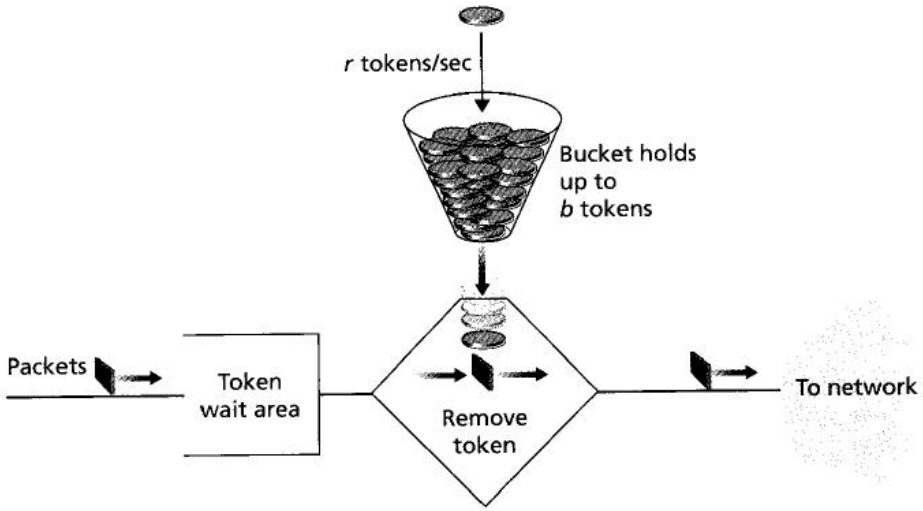
- Two parameters:
  - r: token arrival rate
  - b: bucket depth



- Modus operandi:
  - tokens arrive at the bucket at constant rate r
  - tokens are used by incoming packets
    - a packet of size M uses M tokens when sent
    - if no sufficient tokens exist: packet waits
  - bucket depth bounds number of tokens accumulated
    - when bucket is full: additional tokens are discarded

Token/Leaky Bucket

- Two parameters:
  - r: token arrival rate
  - b: bucket depth



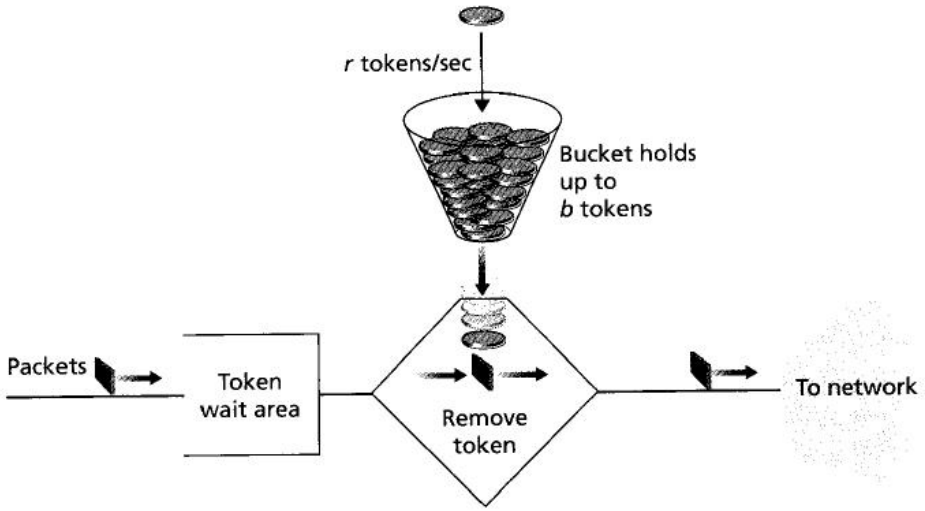
- Properties of token/leaky bucket:
  - amount of data source transmits during interval I is bounded:

$$r\,|I| + b$$

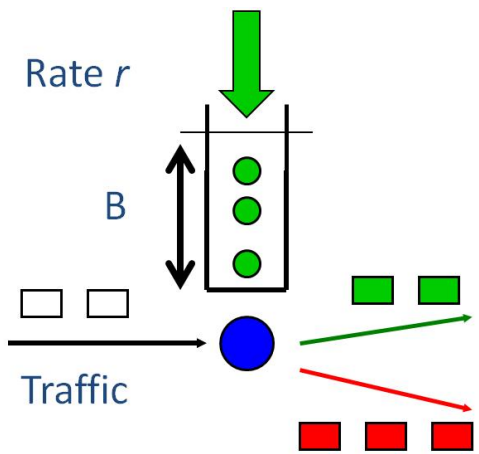
  - long term average rate of traffic: r
  - maximum burst size generated by source: b

Token/Leaky Bucket

- Two parameters:
  - r: token arrival rate
  - b: bucket depth



- Policing and Marking in IntServ
  - Traffic conformance to *TSpec* is monitored at network edge
  - Nonconforming packets:
    - treated as best-effort
    - marked with drop priority



DiffServ - Key Components

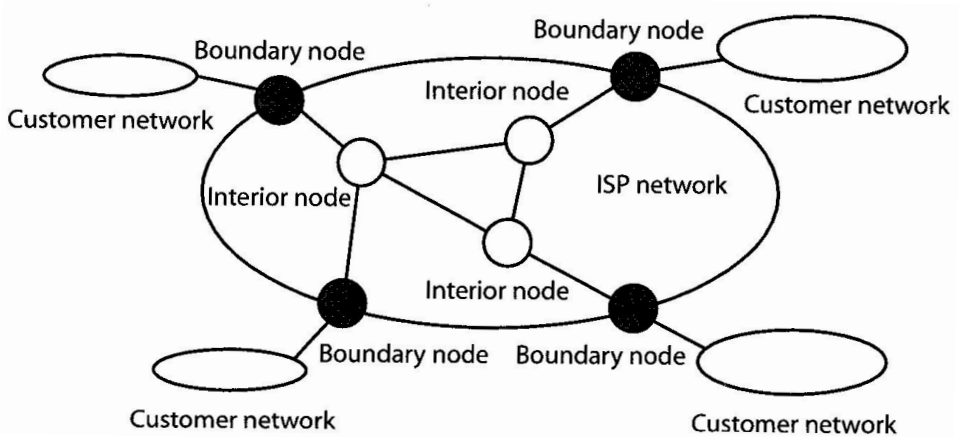
- Main motivation: IntServ handicaps
  - per-flow state
  - complex reservation mechanisms
  - requires large-scale deployment
- Main concern: being "better than best-effort"
- Main design concept: locality of decision
- Key components:
  - middle ground between IntServ and best-effort
  - divide traffic into small number of *forwarding classes*
    - E.g., Gold, Silver, Bronze, Best-Effort
    - class encoded in packet IP header
    - resources allocated per-class (aggregate)
    - IntServ: per-flow resource allocation

DiffServ - Key Components

- Main motivation: IntServ handicaps
  - per-flow state
  - complex reservation mechanisms
  - requires large-scale deployment
- Main concern: being "better than best-effort"
- Main design concept: locality of decision
- Key components:
  - focus on single domain (not E2E)
    - each domain might treat each class differently
    - common "guidelines" among domains
    - allows for incremental deployment:
      non-DiffServ domains are not prohibitive
    - IntServ: non-IntServ node "ruins" reservation

DiffServ - Key Components

- Main motivation: IntServ handicaps
  - per-flow state
  - complex reservation mechanisms
  - requires large-scale deployment
- Main concern: being "better than best-effort"
- Main design concept: locality of decision
- Key components:
  - not all nodes are the same
    - edge-nodes / boundary-nodes $\neq$ core-nodes
    - classification/marking vs. plain forwarding



DiffServ - Key Components

- Main motivation: IntServ handicaps
  - per-flow state
  - complex reservation mechanisms
  - requires large-scale deployment
- Main concern: being "better than best-effort"
- Main design concept: locality of decision
- Key components:
  - based on Service Level Agreements (SLAs)
    - customer-provider long-term contracts
    - IntServ: per-flow reservations
  - mechanisms
    - marking
    - scheduling (priority queuing, WFQ)
    - buffer management (AQM/RED)
  - no deterministic performance guarantees

Services and Forwarding Treatment

- IntServ provides *services*
  - E.g., E2E delay guarantee
- DiffServ provides *forwarding treatments*
  - E.g., express-forwarding

Example

- Express-forwarding treatment
  - two priorities: high and low
  - high always sent before low
  - express-forwarding customers have packets marked "high"
- No-loss service
  - subscriber's packets are never dropped

- Express-forwarding $\neq$ no-loss, but
- Express-forwarding + admission control $\Rightarrow$ no-loss
- FIFO + ? $\Rightarrow$ no-loss

Service Level Agreements (SLAs)

- Specification of expected traffic per-class
  - token-bucket envelope
    - Committed/Peak Information Rate (CIR/PIR)
    - Committed/Peak Burst Size (CBS/PBS)
- Performance metrics provided by the network
- What-if
  - dealing with non-conforming traffic (customer violation)
  - performance degradation (provider violation)
- Additional issues:
  - business-related
  - security, monitoring, accounting, pricing, billing, ...

PHB in a Nutshell

- Recall DiffServ key components:
  - local decisions
  - small number of classes

PHB (Per-Hop Behavior)

Observable behavior of a link for a specific class

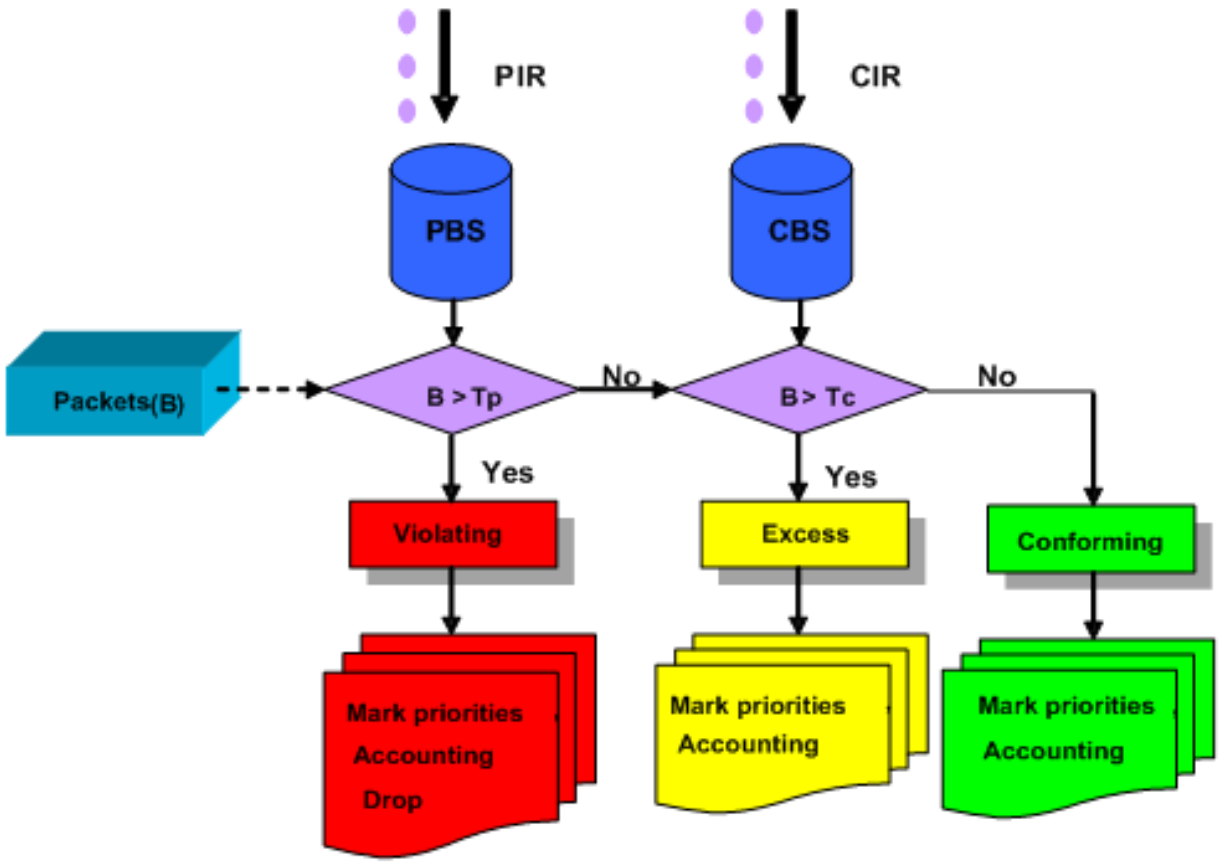
- Observable: bandwidth, delay, loss
- Absolute terms vs. relative terms
- Does not specify *mechanisms*!!

Classification and Conditioning

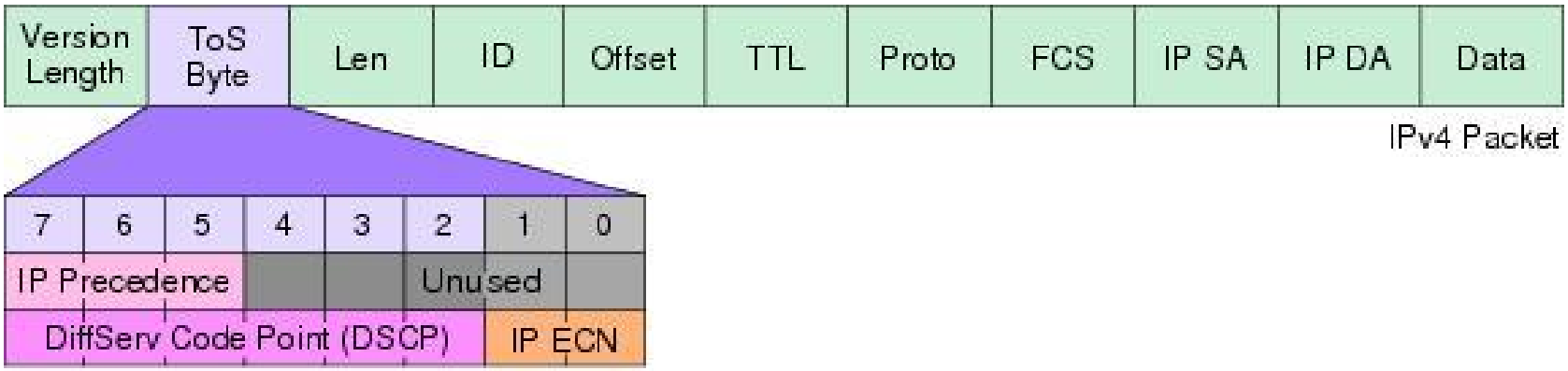
- Done at edge/boundary nodes
- Assigns/reassigns class "code" to packets
- Classification:
  - multifield (MF) vs. behavior aggregate (BA)
    - MF: based on IP header 5-tuple (usually at edge)
    - BA: based on previously assigned code in header
  - based on packet "identity"
- Conditioning and Policing
  - enforce conformance to SLA
  - use token bucket based metering and marking
  - also shaping/dropping
  - based on the overall traffic
- PHB promotion/demotion

Three Color Marking

- Done using a dual token bucket
- E.g., two-rate three color marking



DiffServ Codepoint (DSCP)



- *Class selector* codepoints

$$\boxed{<\text{xxx}000>}$$

  - default PHB: $<000000>$ (best-effort)
  - at least two different forwarding classes
    - 8 DSCPs
    - some might be mapped to the same PHB
  - monotonically non-decreasing (better) priority
  - backward compatibility
    - IP precedence
    - $<110\text{xxx}>/<111\text{xxx}>$: better than other (control/routing)

Assured Forwarding (AF) PHBs

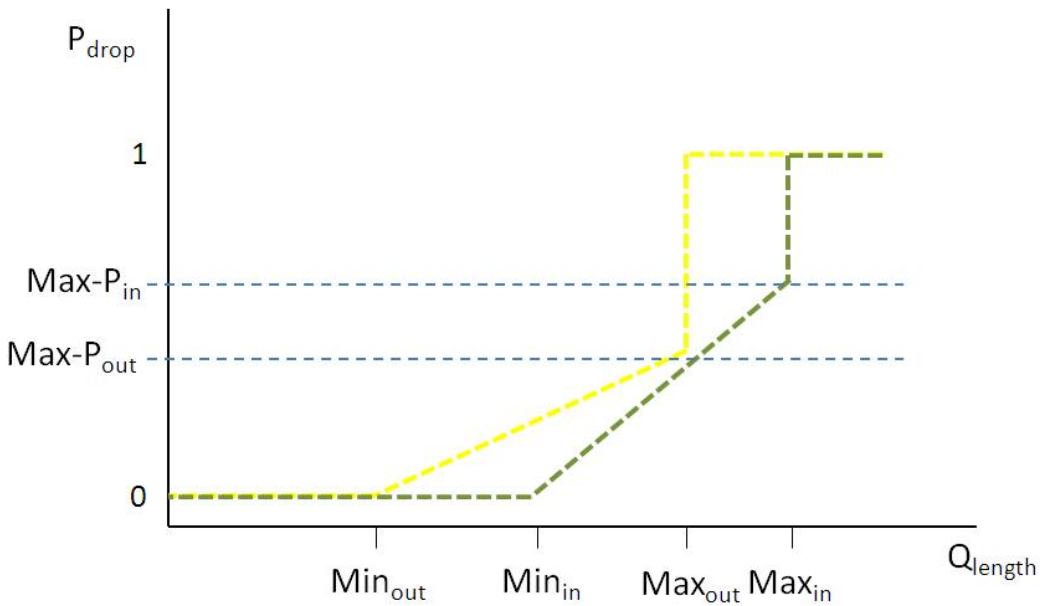
- 4 classes
  - DSCP: <001xx0>, <010xx0>, <011xx0>, <100xx0>
  - per class bandwidth allocation
    - implemented using scheduling (e.g., WFQ)
    - each link can determine its own allocation
- 3 drop priorities per class

DROP Precedence	Class #1	Class #2	Class #3	Class #4
Low Drop Precedence	(AF11) 001010	(AF21) 010010	(AF31) 011010	(AF41) 100010
Medium Drop Precedence	(AF12) 001100	(AF22) 010100	(AF32) 011100	(AF42) 100100
High Drop Precedence	(AF13) 001110	(AF23) 010110	(AF33) 011110	(AF43) 100110

  - "high drop precedence" = more dropping...

Implementing AF

- Multiple drop priorities:
  - usually generalizations of Random Early Detection (RED)
  - RED with In-and-Out (RIO)



  - weighted RED (WRED)
- Queuing/Forwarding:
  - PQ, WRR, variants of WFQ
- Recall: DiffServ does not specify mechanisms!!

Expedited Forwarding (EF) PHB

- DSCP: $<101110>$
- Reserving bandwidth for low-delay low-loss traffic
- EF traffic is queued in high-priority queue
  - e.g., in WFQ/WRR, would get very large weight
- Receives preferential treatment compared to all other PHBs
- Best suited for, e.g., VOIP
- Usually 10-30% of link capacity

DiffServ Deployment

- DiffServ in the Internet backbone
  - very slow deployment
  - VOIP/video seem to work well as-is
  - "cheap" to add bandwidth
- DiffServ outside the backbone
  - common in corporate networks
    - e.g., private VOIP networks use EF
  - residential routers
    - DSL is asymmetric (uplink vs. downlink)
    - upstream priority for VOIP
- Usage of intra-domain local PHBs (non-standardized)
  - unused DSCPs

References

- Kurose and Ross, "*Computer Networking: A Top-Down Approach*", 5th ed., Addison-Wesley, 2010.
- Peterson and Davie, "*Computer Networks: A Systems Approach*", 4th ed., Morgan Kaufmann, 2007.
- Wang, "*Internet QoS: Architectures and Mechanisms for Quality of Service*", Morgan Kaufmann, 2001.
- White, RSVP and Integrated Services in the Internet: A Tutorial. IEEE Communications Magazine, May 1997