CS 5224

Access Control and End-to-end Performance

Dr. Chan Mun Choon
School of Computing, National University of Singapore

Outline

- Policing using Leaky Bucket
- End-to-end Performance using Leaky Bucket and WFQ
- Reference:
  - Mischa Schwartz, "Broadband Integrated Networks," Chapter 4 and 6.

Access Control

Goal: limit traffic to not exceed declared parameters

- Monitor and control the traffic sent by user to network
  - Ensure it conforms to the traffic descriptors specified
  - Users found violating their "agreements" will have packets tagged or dropped
  - Also called Usage Parameter Control (UPC), credit management, traffic "policing"
- Traffic may be "shaped" or "smoothed" to reduce any adverse impact on the network
  - Usually, buffer the packets at the "access" routers and then send out packets at a smoothed, more regular rate

Policing Mechanisms

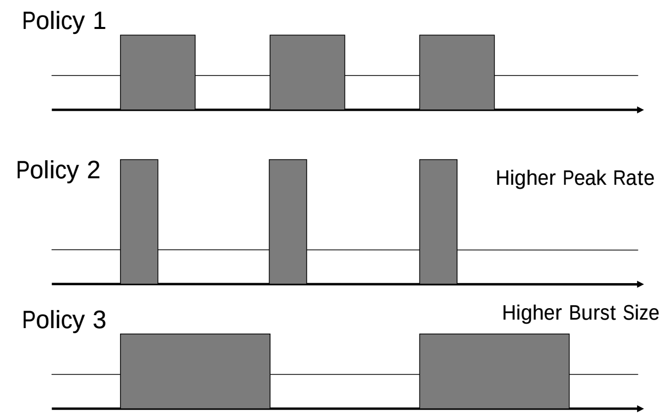
Three common-used criteria:

- *(Long term) Average Rate:* how many packets/bits can be sent per unit time (in the long run)
  - crucial question: what is the interval length: 100 packets per sec or 6000 packets per min have same average!
- *Peak Rate:* e.g., 6000 pkts per min. (ppm) avg.; 15000 ppm peak rate
- *(Max.) Burst Size:* max. number of pkts/bits sent consecutively (with no intervening idle)

Example

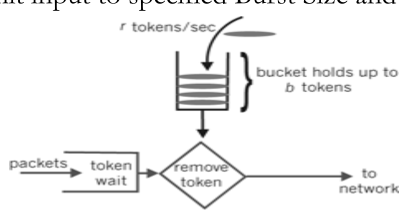
- Policy 1 (assume ATM, constant 53 bytes pkt)
  - Average rate = 1000pps or 424Kbps
  - Peak rate = 2Kpps or 848Kbps
  - Burst Size = 1000 packets or 424Kb
- Policy 2 (assume ATM)
  - Average rate = 1000pps or 424Kbps
  - **Peak rate = 4Kpps or 1696Kbps**
  - Burst Size = 1000 packets or 424Kb
- Policy 3 (assume ATM)
  - Average rate = 1000pps or 424Kbps
  - Peak rate = 2Kpps or 848Kbps
  - **Burst Size = 2000 packets or 828Kb**

Example (Worst Case)



Policing Mechanisms

Token Bucket: limit input to specified Burst Size and Average Rate.



- bucket can hold σ tokens
- tokens generated at rate ρ *token/sec* unless bucket full
- *over interval of length t: number of packets admitted less than or equal to* $(\rho t + \sigma)$.

Deterministic Bound

- If the amount of traffic sent over any time interval can be bounded, can performance bound be provided?
- Yes, a deterministic bound can be provided:
  - Reference:
    - R. Cruz, "A Calculus for Network Delay, Part I and Part II," IEEE Trans on Information Theory, Jan 1991.
    - Parekh and Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Network: The Mulitple Node Case." IEEE/ACM Trans. On Networking, Apr 1994.

Deterministic Bound

- Recall that using leaky bucket policy, over interval of length t, the number of packets admitted less than or equal to $(\rho\, t + \sigma)$.
  - Assume peak rate is "infinity"
- Let GPS be implemented along the routers and g(k) be the service rate allocated at router k, r(k) be the link rate
  - Let g(min) be the smallest rate allocated over all k routers
  - The worst case end-to-end delay (D) is bounded by D < = **σ/g(min)**
- If WFQ is implemented, due to the effect of packet switching
  - D <= **$\sigma/g(min) + \Sigma_{k=1 \text{ to } N-1} Pmax/g(k) + \Sigma_{k=1 \text{ to } N} Pmax/r(k)$**

Example

- Consider a connection with leaky bucket parameters (16KB, 150Kbps). Pmax=8KB. What is g if end-to-end is to be less than 70ms?
  - $\Sigma_{i=1 \text{ to } k} Pmax/r(i)$ = 10 * 8192 * 8/45M = 14.56ms
  - $\sigma/g + (k-1)Pmax/g$ <= 55.44ms
  - g = (16*8*1024 + 9*8192*8)/0.05544 = 13Mbps
- Note that the required rate is 13Mbps/150Kbps = 86.7 times the average rate
  - Large packets can cause substantial delay
  - If packet size is reduce to 1.5KB, g = 3.6Mbps
  - If packet size is further reduced to 53 bytes, g = 289 Kbps

How Efficient?

- GPS/WFQ provides deterministic (wordst case) bounds
  - In reality most packets may not experience close to maximum delay
  - The amount of scheduling resource required is often substantial
- Statistical bounds are much more efficient
  - E.g. < 0.1% of the packets have delay more than 70ms
  - However, statistical bounds (e.g. using equivalent bandwidth) are much harder to compute in practice