TCP Congestion Control

1. How does a sender sense congestion?
   - 

2. How does a sender determine its sending rate?
   - 
   - 

3. What algorithm is used to change the send-rate?
   - Many phases and alternatives...
Reaction to Loss Events

- Exponential increase switches to linear increase when CongWin gets to the ‘threshold’ value (size)
Identify everything on this graph

TCP Congestion Control Algorithm

Increase Sending Rate Phase Options:
1. When CongWin is below Threshold, sender in slow-start phase, window grows exponentially.
2. When CongWin is above Threshold, sender is in congestion-avoidance phase, window grows linearly.

Decrease Sending Rate Phase Options:
1. When a triple duplicate ACK occurs, Threshold set to CongWin/2 and CongWin set to Threshold.
2. When timeout occurs, Threshold set to CongWin/2 and CongWin is set to 1 MSS.
TCP Congestion Control Algorithm

Three major phases / mechanisms:

1) Slow start - at 1 max segment size
   - But increase __________

2) Congestion Avoidance phase
   - AIMD = additive incr, multiplicative decr
   - Using cwnd and ssthresh

3) Fast Recovery
   - Increase of cwnd each round trip time
   - Slow start: __________
   - Congestion avoidance: __________

Summary TCP reaction to loss

- Loss indicated by timeout
  - cwnd set to 1 MSS
  - ssthresh set to cwnd/2
  - Window (cwnd) grows exponentially (slow start) to the threshold, then grows linearly

- Loss indicated by 3 duplicate ACKs
  - Network capable of delivering some segments, so...
  - cwnd is cut in half (=ssthresh)
  - Window grows linearly
Transport Layer Review

- The transport layer services are:
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 

Transport Layer Review

- The transport layer *does not* provide:
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  -
Transport Layer Review

- Compare TCP and UDP (pros and cons?)
- ...
- ...
- ...
- ...
- ...
- ...
- ...
- ...
- ...
- ...

Transport Layer Review

- TCP Connection Management includes
  - ...
  - ...
  - ...
  - ...
  - ...
  - ...
  - ...
  - ...
  - ...
  - ...
  - ...
Transport Layer Review

- Elements of TCP reliability:
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 

Transport Layer Review

- Elements of congestion control algorithm
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
  - 
TCP Flow Control

- URG: urgent data (generally not used)
- ACK: ACK # valid
- PSH: push data now (generally not used)
- RST, SYN, FIN: connection estab (setup, teardown commands)
- Internet checksum (as in UDP)
- Sequence number
- Acknowledgement number
- Head len
- URG data pointer
- Receive window
- Options (variable length)
- Application data (variable length)
- Source port #
- Dest port #
- Checksum
- # of bytes rcvr willing to accept
- Counting by bytes of data (not segments)

TCP flow control (quick & easy)

- **Receiver** “advertises” free buffer space by including rwnd value in TCP header of receiver-to-sender segments
  - **RcvBuffer** size set via socket options (typical default is 4096 bytes)
  - Many operating systems auto-adjust **RcvBuffer**
- **Sender** limits amount of un-ACKed (“in-flight”) data to receiver’s **rwnd** value
  - Guarantees receive buffer will not overflow
**TCP flow control**

The application may remove data from the TCP socket buffers....

... slower than the TCP receiver is delivering it into the buffers (sender is sending)

*flow control to the rescue!*

receiver controls sender, so sender won’t overflow receiver’s buffer by transmitting too much, too fast

---

**Transport Layer Review**

- Other questions?
A fun tangent...
Finite State Machines

Finite State Machines

![Diagram of a finite state machine with states On and Off, and transitions labeled with actions 'Flip switch down' and 'Flip switch up'.]

![Diagram of a finite state machine with states $q_0$, $q_1$, $q_2$, and $q_3$, and transitions labeled with inputs 0 and 1. The start state is $q_0$.]

21

22
TCP sender events:

(1) data received from application:
1. Create a segment and assign a SEQ number
   v SEQ # is byte-stream number of first data byte in segment
2. Start timer if it is not already running
   v Timer is for the oldest un-acked segment
   v Expiration interval: TimeOutInterval

(2) timeout:
1. Retransmit segment that caused the timeout
2. Restart the timer

(3) ACK received:
   q For previously unacked segments
   1. update what is known to be acked
   2. start timer if there are outstanding segments

TCP Congestion Control
TCP Congestion Control: FSM

Transport Layer Summary

- TCP and UDP Services
- Encapsulation (create and attach header)
- Multiplexing and demultiplexing
- Checksum
- Connection management
- Reliable transport service
- Congestion control
- Detect loss and retransmit
  - Detect out-of-order and reorder
- Flow Control
Transport services and protocols

- Provide logical communication, a virtual connection...

...between application processes running on different hosts.

This is not a physical path including routers.

Introduction to the Network Layer

- Desired network layer services...
  - Actual network layer services
- Implemented in hosts and routers
- Two main network layer functions
- Three main network layer protocols
Network Layer Services of IP?

- Guaranteed delivery?
- Guaranteed minimum delay?
- In-order datagram delivery?
- Guaranteed minimum bandwidth to flow?
- Restrictions on changes in inter-packet spacing?

- **IP Provides?** → “Best-effort service”

Key Network-Layer Functions

1. **routing:** determine route taken by packets from source to destination
   - Network-wide routing algorithms

2. **forwarding:** move packets from router's input link to appropriate output link
   - Internal to a single router
Network Layer: Routing and Forwarding

Create versus use the forwarding table

Network Layer, Chapter 4

- Router ‘switching fabric’
  - Hardware / electrical pathways within a router
- Forwarding - use the forwarding table to transmit, or forward, each packet to the correct output link, based on the destination IP address
- Routing - Create the forwarding tables
  - Decentralized vs. Centralized algorithms
  - Within an ISP vs. between ISPs
- Software Defined Networks, SDN