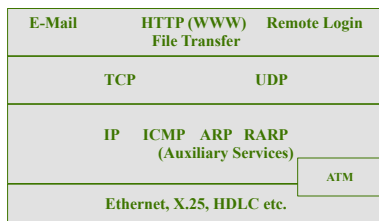


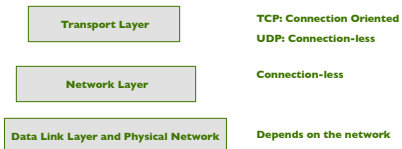
Performance “Just enough TCP”

ITL

Protocol Overview



Connection Types in TCP/IP



What is Performance?

- What data rate can I sustain (on average) between two nodes on the Internet?
 - Throughput
 - “Speed”
- How long does it take to get a reply for a request I send out?
 - Latency
 - Round Trip Time

“Frame Transmission Time”

- Depends on
 - Frame Length
 - Channel Bit Rate

$$t_{frame} = \frac{L}{R}$$

L = Frame Length in Bits

R = Channel Data Rate in Bits per Second

Example: 1500byte packet, 1Mbps link

1500bytes = 12000bits

12000bits/1Mbps = 12msec

“Frame Propagation Time”

- Depends on
 - Physical distance between stations
 - Signal propagation speed
 - 2×10^8 to 3×10^8 meters/second

$$t_{prop} = \frac{d}{v}$$

d = distance between stations in meters

v = speed of signal propagation

5000 miles = 8.05×10^6 m; 8.05×10^6 m/ 3×10^8 m = 27msec

Round Trip Time (RTT)

- Combination of
 - Frame Transmission times on intermediate links
 - Frame propagation times on all links
 - **Queue wait times** in all intermediate nodes
- What does this have to do with speed?
 - Transmission speed is of course limited by link speeds, **but**
 - Restrictions on data senders are imposed as “so many bytes per RTT” - big RTT = slow speed

Congestion

- or, why do we need controls?
 - Lets assume
 - 50Mbps link from an ISP to the Internet
 - 200 customers online
 - 50 packets per customer (1500bytes each) in queue
 - 10,000 packets
 - 15,000,000 bytes
 - 120,000,000 bits
 - **2.4 second queue delay**
- Queue size must be controlled

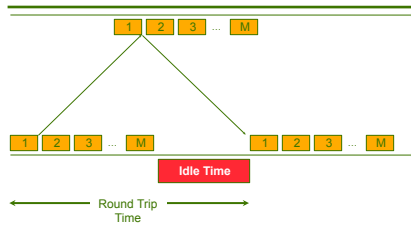
Control Mechanisms

- Connection-Less traffic
 - Packets representing overload are discarded
 - Retransmissions should not take place
 - Applications should adapt
- Connection-Oriented traffic
 - Packets representing overload are also discarded
 - Lost packets must be retransmitted
 - Hosts must adjust sending rate

Sliding Window flow control

- Invented to control flow to slower devices (i.e. end-to-end)
- Almost accidentally provides a cap on data flow rate
- TCP uses a sliding window mechanism for multiple purposes
 - Rate Control
 - Congestion Control/Avoidance
- The window size is dynamically adjusted

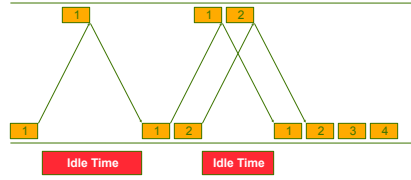
“Sliding Window”



TCP and Congestion Issues

- Slow Start - New Connection
 - Set send window to $n \cdot \text{MSS}$ ($n \leq 4$)
 - MSS = Maximum Segment Size
 - Increase the window by MSS for each ack received
 - Exponential increase in send window size
- What is the limit?
 - Window size reached before full utilization
 - Path is overloaded and an intermediate router discards one or more packets

Slow Start Congestion Control



Note: recent TCP amendments permit more than 1 initial segment

Window doubles in each "cycle"

Window Size and Throughput

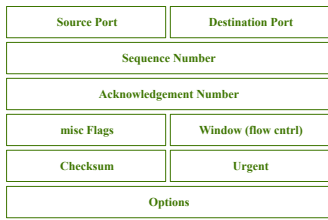
- Maximum throughput is
 - (window size)/RTT
 - Actual throughput may of course be less.
- Bandwidth-Delay-Product
 - $BDP = RTT * (\text{effective bandwidth})$
 - Measures the amount of data "in the network" at any given time.

TCP and UDP

UDP Header

Source Port	Destination Port
Length	Checksum

TCP Header



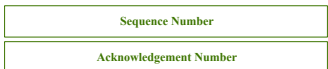
TCP Connection Setup

- "Three-Way Handshake"
 - Send SYN packet
 - Wait for peer to return a SYN/ACK packet
 - Acknowledge the SYN/ACK packet

TCP Connection Termination

- Send a FIN packet
- Wait to receive acknowledgement of FIN

TCP Data Exchange



- Sequence Numbers - Sliding Window
 - Arbitrary initial setting
 - Labels the first byte of the segment
- Acknowledgements
 - Indicate the next byte the receiver is looking for, all previous bytes have been received.

Controls and Measurements

- TCP Window
 - Most controls are automatic
 - Operating systems can and do limit the largest possible TCP window
- Measurements
 - Ping, Traceroute
 - Test tools that imitate real traffic flow
 - iperf (see the assigned reading)