Announcements

- **Reading**
  - Chapter 6 (6.3 & 6.4)
- **Project #3**
  - Is on the web
- **Midterm #1**
  - Last day to request a re-grade is Th 10/18
TCP Protocol

- **TSAPs**
  - Use `<host, port>` combination
  - Well known ports provide services
    - first 256 ports
    - SMTP 25, Telnet 23, Ftp 21, HTTP 80

- **Provides a byte stream**
  - this is **not** a message stream
  - a message (single call to send) may be split, merged, etc.

- **Urgent Data field**
  - provides cut through delivery *within* a transport connection
  - used to send breaks or other high priority info
TCP Packet Format

- Permits ACKs to be piggy packed
  - ACK is next byte expected
  - ACK is only valid if ACK bit is set
- Sequence number
  - first byte in packet
- Also used for connection establishment

<table>
<thead>
<tr>
<th>Source Port</th>
<th>Destination Port</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Sequence Number</td>
<td></td>
</tr>
<tr>
<td>Acknowledgment Number</td>
<td></td>
</tr>
<tr>
<td>Window Size</td>
<td>Urgent Pointer</td>
</tr>
<tr>
<td>Checksum</td>
<td></td>
</tr>
<tr>
<td>0 Or More Options</td>
<td></td>
</tr>
</tbody>
</table>

32 bits
TCP Connection Management

- Three-way Handshake
- Initial Sequence Numbers
  - Use a 4 micro-second clock
  - hosts must wait $T$ (120 seconds) before a reboot
- Connection Closure
  - Each side uses a FIN and FIN_ACK message
  - A FIN times out after $2T$ (240 seconds)
  - Keep alives used to timeout half dead connections
TCP Flow Control

- **Use Variable Sized Sliding Window**
  - ACK indicates start of window
  - Window size indicates current size of window
- **Receiver can send a window of 0**
  - indicates that it want to pause connection
  - urgent data need not follow this request
- **Window size of 16 bits is too small**
  - 64K Bytes
  - only a small fraction of the in-flight bytes when
    - bandwidth is high
    - delay is high
  - solution: window shift option:
    - bit shift window up to 16 bits
    - permits up to $2^{32}$ byte windows
    - reduces window granularity
TCP Congestion Control

- **Detecting Congestion**
  - In general it is difficult
  - But, consider why a packet might be dropped
    - link error - but links are very reliable now
    - buffer overflow --> congestion
  - Use re-transmission timeouts as an estimate of congestion
- **Dealing with Congestion**
  - add a second window (congestion window)
    - limit transmissions to min(recv window, congestion window)
  - start with congestion window = max segment window
    - initial max segment is one kilo-byte
    - on a ACK without a timeout
      - if window < threshold, increment by one max segment
      - otherwise increment by initial max segment
  - on timeout
    - cut threshold in half
    - set window size to initial max segment
TCP Congestion Window

From: Computer Networks, 3rd Ed. by Andrew S. Tanenbaum, (c)1996 Prentice Hall.