Best-Effort Networks and Layering

Lecture 18
November 9, 2009
6.02 Spring 2009

• MAC protocol wrap-up
• Best-effort networks
• Protocol layering

Queues in Packet-Switched Networks
• Queues absorb bursts of arriving packets
  • What happens if queue overflows?
  • Packets are dropped (lost)
  • Can we design a packet-switched network to avoid such losses?
    • Turns out we can’t (or shouldn’t); it would be a bad idea to try!

Packets May Be Dropped in a Packet-Switched Network
• Can’t usually provision link for peak rate
• What if we just buffer all packets?
  • Bad idea: long queues don’t help throughput
  • They only increase delay (from Little’s law)
• So, drop packets when queues overflow
  • Signals congestion
  • At a higher layer, detect and recover from loss
  • Loss not a new symptom

Networks Have Four Kinds of Delays
• Propagation delay
  • Time for one bit to go from sender to receiver
  • Depends on speed of light in communication media
• Transmission delay
  • Time for packet of size S to reach other end
  • Depends on rate, R, of path (delay = S/R)
• Processing delay
  • Time for each switch to process packet (lookup, etc.)
• Queueing delay (variable)

Packets May Be Reordered
• Routing protocol can choose different paths
• Helps recover from link/switch failures
• May help balance load across different links or paths

“A Best-Effort” Network
• Packets may be lost
• Packets may be corrupted
• Packet delays are variable
• Packets may be reordered
• Packets may even be duplicated
• But, it’s relatively easy to be a switch in a network with such a “contract”
• How to find good paths and deliver data in the face of these best-effort properties?
Getting Organized

- Need modularity
  
  \[ A \] \textbf{call} \[ B \]
  
  - A need not know how B is implemented
  - B's implementation can change as long as B's interface doesn't
  - That is, the abstraction provided by B is the only contract between B and A

Layering

- Layering is a particular form of abstraction
  
  - The system is broken into a vertical stack of protocols
    
    - The service provided by one layer is based solely on the service provided by layer below
      
      - This is the “up/down” interface
    
    - Peer interfaces across the network implement communication protocols

Layering in the Internet

<table>
<thead>
<tr>
<th>Application</th>
<th>Transport</th>
<th>Network</th>
<th>Data Link</th>
<th>Physical</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP, FTP, SMTP, BitTorrent, …</td>
<td>TCP, UDP</td>
<td>IP</td>
<td>Ethernet, WiFi, …</td>
<td></td>
</tr>
<tr>
<td>End-to-End Layer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Everything else! Reliability, integrity, ordering, jitter control, …</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Forwarding &amp; routing (and addressing) Framing, (limited) retry, channel access Coding, (De)modulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Layering Interfaces

- Link and network layers are implemented everywhere
- The end-to-end layer (i.e., transport and application) is implemented only at hosts

An Example

- Each layer adds/strips off its own header (and possibly a trailer)
- Each layer may split up higher-level data
- Each layer multiplexes multiple higher layers
- Each layer is (mostly) transparent to higher layers
The Internet “Hourglass”

Applications
- Web
- FTP
- Mail
- News
- Video
- Audio
- ping
- P2P apps

Transport protocols
- TCP
- UDP
- ICMP

Link technologies
- Ethernet
- 802.11
- Power lines
- ATM
- Optical
- Satellite
- Bluetooth

- Many applications, transports, and link protocols
- All use IP at the network layer: universal network layer