CS118 Discussion 1A, Week 3

Zengwen Yuan (zyuan [at] cs.ucla.edu)
Boelter Hall 9436, Friday 12:00—1:50 p.m.
Outline

- Lecture Review: Transport Layer
- Solutions for Homework 2
Transport Layer V.S. Network Layer

- Network layer: logical communication between **hosts**
  - **IP address** is used for identifying a host
- Transport layer: logical communication between **processes**
  - **IP address and port number** are used for identifying a process
Multiplexing and De-multiplexing

- Multiplexing at send host: gather data from multiple sockets
- De-multiplexing at receiving host: deliver received segments to the right socket
- **Five tuples** (src_ip, src_port, dst_ip, dst_port, protocol) are used for multiplexing/demultiplexing
  - How to identify a TCP/UDP socket? *Is*of -i
  - Can TCP and UDP share the same port numbers? **Yes!** e.g. DNS
**UDP**

- No connection establishment
- No connection state
- Small packet overhead (8 byte)
- How to calculate checksum?
  - **Pseudo header** + **UDP header**
    + data
  - Also applicable to TCP
  - Why pseudo header?

![UDP header format diagram]

<table>
<thead>
<tr>
<th>source port</th>
<th>destination port</th>
</tr>
</thead>
<tbody>
<tr>
<td>length</td>
<td>checksum</td>
</tr>
</tbody>
</table>

![UDP header structure diagram]

- source IP address
- destination IP address
- zero
- protocol
- UDP length

**UDP header**

- data
Principles of Reliable Data Transfer

- How to deal with bit errors?
  - Error detection (e.g. checksum)
  - Receiver feedback
  - Retransmission
- Why not error correction?
- How to deal with duplicate packets due to retransmission? **Sequence number**
- How can the sender detect that ACK or data is lost? **Timer**
Stop and Wait Protocol

• Main Issue: **limited performance**

• Consider two hosts that are directly connected by a 50 Kbit/sec satellite link that has a 250 milliseconds propagation delay. If these hosts send 1000 bits segments, what is the maximum throughput in stop-and-wait protocol if we ignore the transmission time of ACK?
Stop and Wait Protocol

• Main Issue: limited performance

• Consider two hosts that are directly connected by a 50 Kbit/sec satellite link that has a 250 milliseconds propagation delay. If these hosts send 1000 bits segments, what is the maximum throughput in stop-and-wait protocol if we ignore the transmission time of ACK?

  • Ans: 1000/(1000/50+250+250)=2 Kbit/sec!
Pipelined Protocols

• Go-back-N: receiver only sends cumulative ACKs
  • Drop out-of-order segments
  • reACK packet with highest in-order sequence number
  • Timer for oldest unACKed packet only, retransmit all unACKed packets

• Selective repeat: receiver ACKs individual packets
  • Buffer out of order segments
  • Timer for each individual unACKed packet, retransmit any unACKed packet
Demo: Selective Repeat/Go Back N


Selecting Repeat / Go Back N

**configuration**
- protocol
  - Go back N
  - Selective Repeat
- window size
  - 5
- end to end delay
  - 5000
- scroll mode
  - Typewriter style
- automatic emission of packets
  - stop

**legend**
- no data received yet
- data buffered (ready to send, delivered or sent but no ack received yet)
- ack
- transmission confirmed
- data has been delivered to upper network layer

**number of packets emitted per minute**
- 50
- 11000

**timeout**
- 11000

The number of packets the upper layer tries to send per minute.

*coded by Johannes Kessler 2012*
TCP

- Point-to-point, byte-stream reliable transport protocol

- **Multiplexing/de-multiplexing**: Source/Dest port

- **Reliable data transfer**: sequence number, ack, checksum, RTT estimation

- **Connection setup**: sequence number, SYN, receive window

- **Connection teardown**: sequence number, FIN

![TCP packet diagram]

- 32 bits
- | source port # | dest port # |
- | sequence number |
- | acknowledgement number |
- | receiver window size |
- | checksum |
- | ptr to urgent data |
- Options (variable length)

Application Data
TCP Connection Management

• Connection setup: three-way handshaking
  • 1st round: SYN+initial sequence number
  • 2nd round: SYN+SYN-ACK+server’s initial sequence number
  • 3rd round: SYN-ACK+ACK+(optional)data

• Connection Teardown
  • “Half-closed” connection
  • Why timed wait?
## Comparison of Reliable Transport Protocol

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Buffer at sender</th>
<th>Buffer at receiver</th>
<th>ACK</th>
<th>Timeout/Retransmission</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop &amp; Wait</td>
<td>No</td>
<td>No</td>
<td>No out-of-order</td>
<td>Retransmit timeout packet</td>
</tr>
<tr>
<td>Go-Back-N</td>
<td>Yes</td>
<td>No</td>
<td>Accumulative Seq#</td>
<td>Retransmit all packets in window</td>
</tr>
<tr>
<td>Selective Repeat</td>
<td>Yes</td>
<td>Yes</td>
<td>Received Seq#</td>
<td>Retransmit timeout packet</td>
</tr>
<tr>
<td>TCP</td>
<td>Yes</td>
<td>Yes</td>
<td>Next expected Seq#</td>
<td>Retransmit timeout packet</td>
</tr>
</tbody>
</table>