Midterm II Review

Bryan Rombach (bmr3ud), Michael Benos (mtb9ps)

March 24, 2020

1 Logistics

- Online through Collab
- 48 hours starting at 12am on Thursday, March 26
- Open slides, notes, book. No internet, no neighbor. Python standard library for `socket` is OK.
- Designed to be 4 hours tops, can be completed in 1:20.
- 10 questions. Focus on the Transport layer and HTTP protocol
  - 4-5 questions will concern a Wireshark trace.
  - 1-2 multiple choice
  - 1-2 short answer calculation
  - A short coding question at the end – blank file, no skeleton code
    - Code is real Python code which will be tested.
    - Ex: ”Write an app program that uses socket create a half duplex text messaging”
  - Questions are not all weighted the same

2 Content to Review

- Socket programming
- Reliable transport protocols and state machines
- Calculating utilization *
- Effect windowing has on utilization
- Pipe-lining
- On/Off buffers - how to know when to send halt signal? *

- Estimating Round Trip Time *
  - Exponentially weighted times
- Analyzing a TCP trace using Wireshark will be taught as part of the exam since this is a new skill
- No state machines concepts, as they were very well covered in homeworks

* Topics which may require numerical calculations

3 TCP and UDP – Brief Review

- Why are transport protocols needed?
  - Queuing losses – when buffers overflow, packets can be dropped
  - Corruption of packets – interference or changes in the physical media
- What is the use case for UDP given that the protocol is unreliable?
  - No overhead - clients and servers don't have to keep track of tracking which packets have been received
  - A good use case for UDP is sending live video or audio over a network since if a few packets are lost here and there, it's not a big problem. Waiting for them to be re-transmitted would make the stream slower and choppier.
- What are the benefits and drawbacks of TCP
  - Order of packets can be maintained using sequence numbers
  - Reliable Transport Protocol – re-sends dropped or corrupted packets using ACKs
  - Selective repetition of packets rather than re-sending en masse

More details for this topic can be found in the Feb. 25 Lecture (”Transport Layer 1”).

4 Flow Control – Brief Review

- A receiver may not be able to handle packets as fast as sender sends them
- Receiver must be able to signal to the sender to stop sending packets for a time
- Thus, we must consider the amount of time it takes for the off signal to propagate

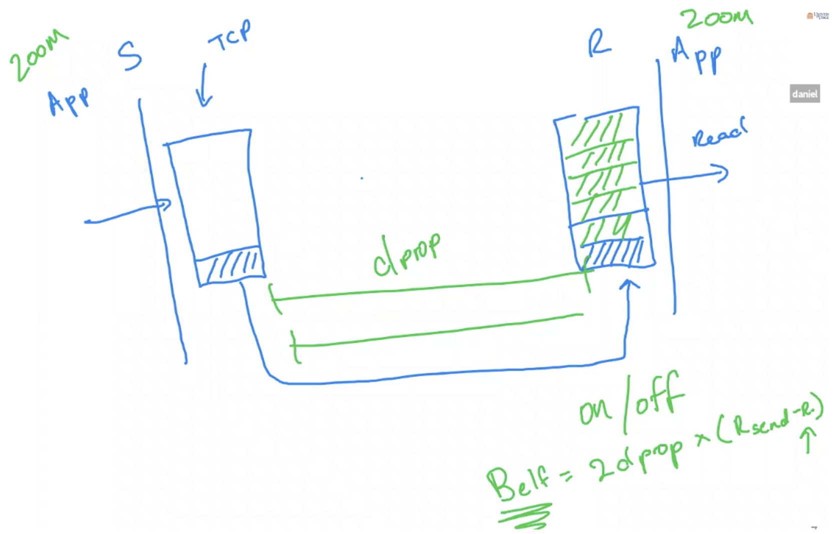

Figure 1: If the receiver gets packets from the sender more quickly than it is able to deal with, a signal must be sent to stop transmission. In this example, the receiver sends the buffer left or B_left. The delay it takes for the signal to propagate (d_prop) must be taken into account.

More details on this topic can be found in the Feb. 27 and Mar. 3 Lectures on the transport layer.

5 Wireshark – Focus Points

- Analyze a TCP trace in the context of the content we learned.
- Know how to identify the three way handshake
  - Look for the SYN → SYN-ACK → ACK exchange between the two actors
  - Each actor must complete at least two actions
  - After each is done transmitting a FIN message is sent and must be ACK'd
- Pay special attention to how sequence and acknowledgement numbers relate