ICSI 516 Computer Communication Networks, Fall 2018
Homework 3 – Transport Layer
Due October 20th, 2018 11:59PM via Blackboard – (20 points)

Requirements and policies

Students should complete this assignment individually. Your turn-in is due on Blackboard. **Late submissions will not be accepted.**

Part 1: Fundamentals of the Transport Layer

1. [2*points*] Similar to the Network layer, UDP provides best effort packet delivery service at the Transport layer. Briefly discuss why can't we simply connect the application layer directly to the network layer if our application does not require reliable transport.

2. [3*points*] Suppose that a FTP server runs in Host C on port 21 (control) and 20 (data). Suppose this FTP server is currently receiving requests from two different Hosts, A and B. Are all of the requests being sent through the same socket at Host C? If they are being passed through different sockets, do all sockets have port 21/20? Discuss and explain.

3. [2*points*] Consider the basic operations of TCP reliability. Which parts of it resemble Go-Back-N and which resemble Selective Repeat. Briefly discuss how TCP overcomes inherent problems in the classical Go-Back-N and Selective Repeat schemes.

4. [1*points*] Why does TCP implement congestion control if it already has flow control to manage the sender window?

5. [1*points*] We developed our realistic reliable data transfer (rdt3.0) protocol by assuming increasingly complicating circumstances (e.g. channel with error and loss) that required additional parameters to handle these circumstances. Why did we need to introduce sequence numbers? How about timers?

6. [1*points*] How does congestion manifest itself in computer communication networks?

7. [*4points*] In this question we will apply our knowledge of the operations of TCP Reno in order to study its behavior through a real-world measurement. Figure 1 shows the congestion window of the protocol over multiple transmission rounds. Study the graph and answer the following questions:

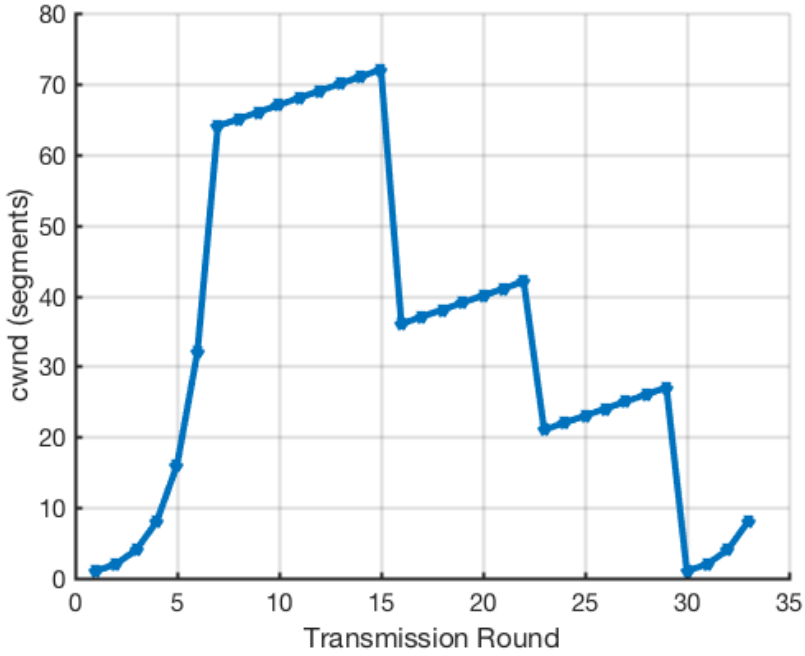

Figure 1: Evolution of TCP congestion window over 33 transmission rounds.

(a) In which round intervals is slow start operating?

(b) Which congestion avoidance mechanism can be observed here?

(c) In which intervals is congestion avoidance operating?

(d) What caused the cwnd decrease in round 16: a triple duplicate ACK or a timeout?

(e) What is the value of ssthresh in round 1?

(f) What is the value of ssthresh in round 18?

(g) In which transmission round is the 70th segment sent?

(h) Imagine that the sender received a triple duplicate ACK after the 26th round. What will be the values of cwnd and ssthresh following this event?

8. [3*points*] Download the following Wireshark trace, load it onto your computer, examine it and answer the following questions:
`http://www.cs.albany.edu/~mariya/courses/csi516S18/hw/tcp-wireshark-trace-1`

   (a) What is the IP address and TCP port number used by the client computer (source) to transfer the file to `gaia.cs.umass.edu`? What is the IP address and port number used by `gaia.cs.umass.edu` to receive the file?

   (b) What is the sequence number of the `TCP SYN` segment that is used to initiate the TCP connection between the client computer and `gaia.cs.umass.edu`? What is it in the segment that identifies the segment as a SYN segment?

   (c) What is the sequence number of the `SYNACK` segment sent by `gaia.cs.umass.edu` to the client computer in reply to the `SYN`? What is the value of the ACKnowledgement field in the `SYNACK` segment? How did `gaia.cs.umass.edu` determine that value?

   (d) What is the sequence number of the TCP segment containing the `HTTP POST` command? Note that in order to find the `POST` command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.

   (e) Consider the TCP segment containing the `HTTP POST` as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the `HTTP POST`)? At what time was each segment sent? When was the `ACK` for each segment received? Given the difference between when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments? What is the `EstimatedRTT` value after the receipt of each `ACK`? Assume that the value of the `EstimatedRTT` is equal to the measured RTT for the first segment, and then is computed using the `EstimatedRTT` equation for all subsequent segments. **Note:** Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the "listing of captured packets" window that is being sent from the client to the `gaia.cs.umass.edu` server. Then select: *Statistics→TCP Stream Graph → Round Trip Time Graph.*

Part 2: Research Papers

1. [3*points*] Consider our recent reading [Chiu+89] Analysis of the Increase and Decrease algorithms for congestion avoidance in computer networks. How does this article showcases that TCP is fair?