CS335 Computer Networks (Fall 2020)

Assignment Three (Due on Oct 23, 2020)

NOTE: *All assignments are to be submitted to URCourses. Please note that the due time of each assignment is at 10:00 pm (URCourses time) on the due date. Please make sure to "save the changes" after uploading your files. You must delete the uploaded file if you want to upload a new version. All uploaded files will be submitted automatically after the deadline and your update time is your submission time. You will be unable to change your files after deadline.*

You may see partial markings from URCourses. Please do not complaint the marking until the announcement of completion marking. You should inform me any errors within 7 days of the announcement. Any request made after that will not be considered.

1. Read Chapters 3, 4 and 8. *(0 point)*

2. UDP and TCP use 1's complement for their checksums. Suppose that you have the following three 8-bit bytes: 01010011, 01100110, 01110100. What is the 1s complement of the sum of these 8-bit bytes? (Note that although UDP and TCP use 16-bit words in computing the checksum, for this problem you are being asked to consider 8-bit sum.) Show all work. Why is it that UDP takes the 1s complement of the sum: that is, why not just use the sum? With the 1s complement scheme, how does the receiver detect errors? Is it possible that a 1-bit error will go undetected? How about a 2-bit error? (*10 points*)

3. Consider the GBN protocol with a sender window of size of 4 and a sequence number range of 1,024. Suppose that the at time t, the next in-order packet that the receiver is expecting has a sequence number of k. Assume that the medium does not reorder messages. Answer the following questions:
    a. What are the possible sets of sequence numbers inside the sender's window at time t? Justify your answer.
    b. What are all possible values of the ACK field in all possible messages currently propagating back to the sender at time t? Justify your answer. (*15 points*)

4. Hosts A and B are communicating over a TCP connection, and Host B has already received from A all bytes up through byte 126. Suppose Host A then sends two segments to Host B back-to-back. The first and second segments contain 80 and 40 bytes of data, respectively. In the first segment, the sequence number is 127, the source port number is 302, and the destination port number is 80. Host B sends an acknowledgement whenever it receives a segment from Host A.
    a. In the second segment sent from Host A to B, what are the sequence number, source port number, and destination port number?
    b. If the first segment arrives before the second segment, in the acknowledgement of the first arriving segment, what is the acknowledgement number, the source port number, and the destination port number?
    c. If the second segment arrives before the first segment, in the acknowledgement of the first arriving segment, what is the acknowledgement number?

d. Suppose the two segments sent by A arrive in order at B. The first acknowledgement is lost and the second acknowledgement arrives after the first timeout interval. Draw a timing diagram, showing these segments and all other segments and acknowledgements sent. (Assume that there is no additional packet loss.) For each segment in your figure, provide the sequence number and the number of bytes of data; for each acknowledgement that you add, provide the acknowledgement number.

(*20 points*)

5. Suppose the five measured SampleRTT values are 106ms, 120ms, 140ms, 90ms, and 115ms. Compute the EstimatedRTT after each of these SampleRTT values is obtained, using a value of α=0.125 and assuming that the value of EstimatedRTT was 100ms just before the first of these five samples were obtained. Compute also the DevRTT after each sample is obtained, assuming a value of β=0.25 and assuming the value of DevRTT was 5 ms just before the first of these five samples was obtained. Last compute the TCP TimeoutInterval after each of these samples is obtained. (*15 points*)

6. In this problem, we consider the delay introduced by the TCP slow-start phase. Consider a client and a Web server directly connected by one link of rate R. Suppose the client wants to retrieve an object whose size is exactly equal to 15 S, where S is the maximum segment size (MSS). Denote the round trip time between client and server as RTT (assumed to be constant. Ignoring protocol headers, determine the time to retrieve the object (including TCP connection establishment) when:
   a. $4S/R > S/R + RTT > 2S/R$
   b. $S/R + RTT > 4S/R$
   c. $S/R > RTT$

   (*15 points*)

7. Create some review questions in the format of multiple choice, true or false, concept matching, filling blanks, etc. that are suitable for quizzes. (*option, bonus up to 5 points*)

8. Practise with Wireshark labs (*0 point*)