ENSC-371 – Summer 2017 Assignment#3: Due Date 04 July 2017

Note: Provide Detailed solutions and not just final answers. Final Answers only will get a mark of zero.

Problem-1:[14]

UDP and TCP use 1s complement for their checksums. Suppose you have the following three 8-bit bytes: 01010011, 01100110, and 01110100.

a. [6] 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 sums.) Show all work.
b. [2] Why is it that UDP takes the 1's complement of the sum; that is, why not just use the sum?
c. [2] With the 1s complement scheme, how does the receiver detect errors?
d. [2] Is it possible that a 1-bit error will go undetected?
e. [2] How about a 2-bit error?

Solution:

a. Note, wrap around if overflow.

$$\begin{array}{r} 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1 \\ +\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 0 \\ \hline 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1 \end{array} \quad \text{and then} \quad \begin{array}{r} 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1 \\ +\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0 \\ \hline 0\ 0\ 1\ 0\ 1\ 1\ 1\ 0 \end{array}$$

One's complement = 1 1 0 1 0 0 0 1.

b. To detect errors, the receiver adds the four words (the three original words and the checksum).

c. If the sum contains a zero, the receiver knows there has been an error.

d. All one-bit errors will be detected.

e. Two-bit errors can be undetected (e.g., if the last digit of the first word is converted to a 0 and the last digit of the second word is converted to a 1).

Note: Provide Detailed solutions and not just final answers. Final Answers only will get a mark of zero.

Problem-2:[6]

Consider the cross-country example shown in Figure 3.17. How big would the window size have to be for the channel utilization to be greater than 98 percent? Suppose that the size of a packet is 1,500 bytes, including both header fields and data.

Solution:

It takes 12 microseconds (or 0.012 milliseconds) to send a packet, as 1500*8/109=12 microseconds. In order for the sender to be busy 98 percent of the time, we must have $util = 0.98 = (0.012n)/30.012$ or n approximately 2451 packets.

Note: Provide Detailed solutions and not just final answers. Final Answers only will get a mark of zero.

Problem-3:[10]

Consider the GBN protocol with a sender window size of 4 and a sequence number range of 1,024. Suppose that 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. [5] What are the possible sets of sequence numbers inside the sender's window at time t? Justify your answer.
b. [5] 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.

Solution:

a. Here we have a window size of N=3. Suppose the receiver has received packet k-1, and has ACKed that and all other preceding packets. If all of these ACK's have been received by sender, then sender's window is [k, k+N-1]. Suppose next that none of the ACKs have been received at the sender. In this second case, the sender's window contains k-1 and the N packets up to and including k-1. The sender's window is thus [k-N,k-1]. By these arguments, the senders window is of size 3 and begins somewhere in the range [k-N,k].

b. If the receiver is waiting for packet k, then it has received (and ACKed) packet k-1 and the N-1 packets before that. If none of those N ACKs have been yet received by the sender, then ACK messages with values of [k-N,k-1] may still be propagating back. Because the sender has sent packets [k-N, k-1], it must be the case that the sender has already received an ACK for k-N-1. Once the receiver has sent an ACK for k-N-1 it will never send an ACK that is less that k-N-1. Thus the range of in-flight ACK values can range from k-N-1 to k-1.

Note: Provide Detailed solutions and not just final answers. Final Answers only will get a mark of zero.

Problem-4:[10]

Consider transferring an enormous file of L bytes from Host A to Host B. Assume an MSS of 536 bytes.

a. [5] What is the maximum value of L such that TCP sequence numbers are not exhausted? Recall that the TCP sequence number field has 4 bytes.

b. [5] For the L you obtain in (a), find how long it takes to transmit the file. Assume that a total of 66 bytes of transport, network, and data-link header are added to each segment before the resulting packet is sent out over a 155 Mbps link. Ignore flow control and congestion control so A can pump out the segments back to back and continuously.

Solution:

There are $2^{32} = 4{,}294{,}967{,}296$ possible sequence numbers.

a) The sequence number does not increment by one with each segment. Rather, it increments by the number of bytes of data sent. So the size of the MSS is irrelevant -- the maximum size file that can be sent from A to B is simply the number of bytes representable by $2^{32} \approx 4.19\,\text{Gbytes}$.

b) The number of segments is $\left\lceil \frac{2^{32}}{536} \right\rceil = 8{,}012{,}999$. 66 bytes of header get added to each segment giving a total of 528,857,934 bytes of header. The total number of bytes transmitted is $2^{32} + 528{,}857{,}934 = 4.824 \times 10^{9}$ bytes.
Thus it would take 249 seconds to transmit the file over a 155~Mbps link.

Note: Provide Detailed solutions and not just final answers. Final Answers only will get a mark of zero.

Problem-5:[16]

Host 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 acknowledgment whenever it receives a segment from Host A.

a. [4] In the second segment sent from Host A to B, what are the sequence number, source port number, and destination port number?
b. [4] If the first segment arrives before the second segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number, the source port number, and the destination port number?
c. [4] If the second segment arrives before the first segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number?
d. [4] Suppose the two segments sent by A arrive in order at B. The first acknowledgment is lost and the second acknowledgment arrives after the first timeout interval. Draw a timing diagram, showing these segments and all other segments and acknowledgments sent. (Assume 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 acknowledgment that you add, provide the acknowledgment number.

Solution:

a) In the second segment from Host A to B, the sequence number is 207, source port number is 302 and destination port number is 80.

b) If the first segment arrives before the second, in the acknowledgement of the first arriving segment, the acknowledgement number is 207, the source port number is 80 and the destination port number is 302.

c) If the second segment arrives before the first segment, in the acknowledgement of the first arriving segment, the acknowledgement number is 127, indicating that it is still waiting for bytes 127 and onwards.

Note: Provide Detailed solutions and not just final answers. Final Answers only will get a mark of zero.

d)