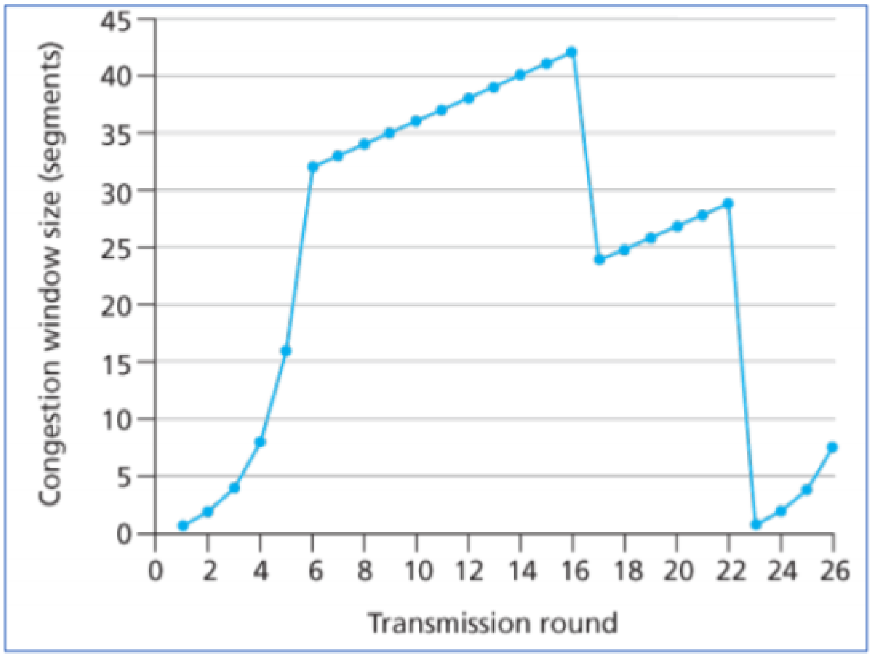
Ocean Lu
CS 3800
Professor Korah
11.09.19

Assignment #4

1. (Ch 3: P40) [4pts] Consider the following figure. Assuming TCP is the protocol experiencing the behavior shown above, answer the following questions. In all cases, you should provide a short discussion justifying your answer.



   a. Identify the intervals of time when TCP slow start is operating.
   At the transmission round from 5-6, it is exponentially fast, which is a nature of a slow start, so the interval [1,6] is a slow start. There is segment loss detected after round 16 and 22, and from the interval 23-26, it slowly increases exponentially. Thus, the operating interval of slow start occurred two times, at [1,6] and [23,26]

   b. Identify the intervals of time when TCP congestion avoidance is operating.
   When there is no congestion in TCP, then with each increase in rounds there is a linear increment in the window size. TCP congestion avoidance is operating time intervals are [7,16] and [17,22]

   c. After the 16th transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?
   Segment loss can be detected based on the congestion window size after the loss. If the congestion window size is reduced but not dropped to 1 after the loss, then the loss is indicated by triple-duplicate-ACK. The interval of loss is 2, so segment

loss detected after 16^{th} transmission round is indicated by triple-duplicate-ACK.

d. After the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?
Segment loss can be detected based on the congestion window size after the loss. If the congestion window size is dropped to 1 after the segment loss, then it is indicated by timeout. The interval of loss is 1, so the segment loss detected after the 22^{nd} transmission round is indicated by timeout.

e. What is the initial value of ssthresh at the first transmission round?
The initial threshold (ssthresh) value is the congestion window size at the point from which the congestion avoid begins. The initial threshold (ssthresh) value at the first transmission round is 32.

f. What is the value of ssthresh at the 18th transmission round?
Segment loss occurred at the 16^{th} round, and so the threshold value is set to half of its size before segment loss. The value before loss is 42, and so $\frac{42}{2} = 21$. The threshold (ssthresh) value during the 18^{th} transmission round is 21.

g. What is the value of ssthresh at the 24th transmission round?
Segment loss occurred at the 22^{nd} round (before 24^{th}), and so the threshole value is set to half of its size before segment loss. $\frac{29}{2} = 14.5 \approx 14$. The threshold value during the 24^{th} transmission is 14.

h. Assuming a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, what will be the values of the congestion window size and of *ssthresh*?
The triple duplicate ACK is received at round 26, which indicate the segment loss. The threshold value and the congestion size, which was 8, are set to half when the loss is identified. Thus, the new threshold value will be 4 and the congestion window size is 7.

2. (Ch 3: P31) [4pts] Suppose that the five measured SampleRTT values (see Section 3.5.3 in textbook) are 106 ms, 120ms, 140 ms, 90 ms, and 115 ms. 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 100 ms 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.

$Five\ SampleRRT: 106\ ms, 120\ ms, 140\ ms, 90\ ms, 115\ ms.$
$First\ Estimated\ RRT\ Value = 100\ ms.$
$\alpha = 0.125, \beta = 0.25$
$First\ DevRTT\ = 5\ ms$

$$EstimatedRTT = (1-\alpha) * EstimatedRTT + \alpha * SampleRTT$$
$$DevRTT = (1-\beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT|$$
$$TimeoutInterval = EstimatedRTT + 4 * DevRTT$$

SampleRTT value 106 ms:

$$EstimatedRTT_{106} = (1-\alpha) * EstimatedRTT + \alpha * SampleRTT$$
$$= (1-0.125) * 100 + 0.125 * 106 = 0.875 * 100 + 0.125 * 106$$
$$= \mathbf{100.75}$$
$$DevRTT_{106} = (1-\beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT|$$
$$= (1-0.25) * 5 + 0.25 * |106 - 100.75| = 0.75 * 5 + 0.25 * 5.25$$
$$= \mathbf{5.06}$$
$$TimeoutInterval_{106} = EstimatedRTT + 4 * DevRTT$$
$$= 100.75 + 4 * 5.06 = 100.75 + 20.24$$
$$= \mathbf{120.99}$$

SampleRTT value 120 ms:

$$EstimatedRTT_{120} = (1-\alpha) * EstimatedRTT + \alpha * SampleRTT$$
$$= (1-0.125) * 100.75 + 0.125 * 120 = 88.15 + 15$$
$$= \mathbf{103.15}$$
$$DevRTT_{120} = (1-\beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT|$$
$$= (1-0.25) * 5.06 + 0.25 * |120 - 103.15| = 0.75 * 5.06 + 0.25 * 16.85$$
$$= \mathbf{8}$$
$$TimeoutInterval_{120} = EstimatedRTT + 4 * DevRTT$$
$$= \ 103.15 + 4 * 8 = 103.15 + 32$$
$$= \mathbf{135.15}$$

SampleRTT value 140 ms:

$$EstimatedRTT_{140} = (1-\alpha) * EstimatedRTT + \alpha * SampleRTT$$
$$= (1-0.125) * 103.15 + 0.125 * 140 = 0.875 * 1 - 3.15 + 0.125 * 140$$
$$= \mathbf{107.756}$$
$$DevRTT_{140} = (1-\beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT|$$
$$= (1-0.25) * 8 + 0.25 * 32.244 = 6 + 8.061$$
$$= \mathbf{14.061}$$
$$TimeoutInterval_{140} = EstimatedRTT + 4 * DevRTT$$
$$= \ 107.756 + 4 * 14.061 = 107.756 + 56.244$$
$$= \mathbf{164.0000}$$

SampleRTT value 90 ms:

$$EstimatedRTT_{90} = (1-\alpha) * EstimatedRTT + \alpha * SampleRTT$$
$$= (1-0.125) * 107.756 + 0.125 * 90 = 0.875 * 107.756 + 0.125 * 90$$
$$= \mathbf{105.536}$$
$$DevRTT_{90} = (1-\beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT|$$

$= (1 - 0.25) * 107.756 + 0.125 * 90 = 0.875 * 107.756 + 0.125 * 90$
$= \mathbf{105.536}$
$TimeoutInterval_{90} = EstimatedRTT + 4 * DevRTT$
$= 105.536 + 4 * 14.43 = 105.536 + 57.72$
$= \mathbf{163.256}$

SampleRTT value 115 ms:
$EstimatedRTT_{115} = (1 - \alpha) * EstimatedRTT + \alpha * SampleRTT$
$= (1 - 0.125) * 105.536 + 0.125 * 115 = 0.875 * 105.536 + 0.125 * 115$
$= \mathbf{106.72}$
$DevRTT_{115} = (1 - \beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT|$
$= (1 - 0.25) * 14.43 + 0.25 * |115 - 106.72| = 0.75 * 14.4 + 0.25 * |115 - 106.72|$
$= \mathbf{12.89}$
$TimeoutInterval_{115} = EstimatedRTT + 4 * DevRTT$
$= 106.72 + 4 * 12.89 = 106.72 + 51.56$
$= \mathbf{158.28}$

3. (Ch 3: P27) [4pts] 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.

   Host B has already received all byptes up through byte 126 from A
   Sequence number of first segment = 127
   Source port number of first segment = 302
   Destination port number of first segment = 80

   a. In the second segment sent from Host A to B, what are the sequence number, source port number, and destination port number?

      Sequence number = sequence number of first segment + number of bytes of data in first segment = 127+80 = 207
      Source port number = 302
      Destination port number = 80

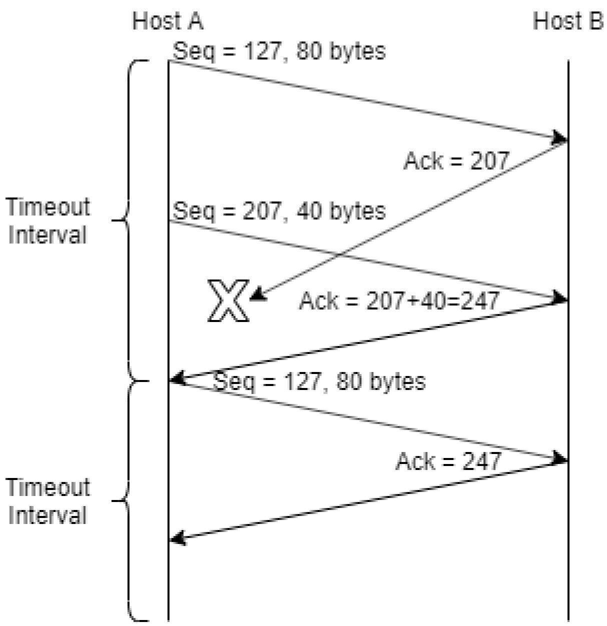
   b. If the first segment arrives before the second segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number, the source port number, and the destination port number?

      Acknowledgement number = 207
      Source number = 80
      Destination port number = 302

   c. If the second segment arrives before the first segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number?

Because the receiver indicates that it is waiting for 127 bytes and upcoming data, the acknowledgement number = 127

d. 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.



4. (Ch 4: P5) [4pts] Consider a datagram network using 32-bit host addresses. Suppose a router has four links, numbered 0 through 3, and packets are to be forwarded to the link interfaces as follows:

Destination Address Range	Link Interface
11100000 00000000 00000000 00000000 through 11100000 00111111 11111111 11111111	0
11100000 01000000 00000000 00000000 through 11100000 01000000 11111111 11111111	1
11100000 01000001 00000000 00000000 through 11100001 01111111 11111111 11111111	2
otherwise	3

a. Provide a forwarding table that has five entries, uses longest prefix matching, and forwards packets to the correct link interfaces.

Prefix Match	Link Interface
11100000 00	0
11100000 01000000	1
11100000 01000001	2
1110001	2
Otherwise	3

b. Describe how your forwarding table determines the appropriate link interface for datagrams with destination addresses:
11001000 10010001 01010001 01010101
11100001 01000000 11000011 00111100
11100001 10000000 00010001 01110111

Address	Link Interface
11001000 10010001 01010001 01010101	Otherwise
11100001 01000000 1100001 00111100	2
11100001 10000000 00010001 01110111	2

5. (Ch 4: P8) [Extra credit 4pts] Consider a router that interconnects three subnets: Subnet 1, Subnet 2, and Subnet 3. Suppose all of the interfaces in each of these three subnets are required to have the prefix 223.1.17/24. Also suppose that Subnet 1 is required to support at least 60 interfaces, Subnet 2 isto support at least 90 interfaces, and Subnet 3 is to support at least 12 interfaces. Provide three network addresses (of the form a.b.c.d/x) that satisfy these constraints.
Subnets can have IP addresses of the form 223.1.17.XXXX/Y.
Subnet 2 has at least 90 interfaces, so 7 bits can support up to 128 addresses. The host addresses range from 0 to 127, which is 00000000 to 01111111. The address range varies by the last 7 bits out of all 32 bits, so the prefix range is $32 - 7 = 25$. Subnet 2 consists of 128 addresses from 223.1.17.0/25 to 223.1.17.127/25.
Subnet 1 has at least 60 interfaces, so 6 bits can support up to 64 addresses. The host addresses range from 128 to 191, which is 10000000 to 10111111. The address range varies by the last 6 bits out of all 32 bits, so the prefix range is $32 - 6 = 26$. Subnet 1 consists of 64 addresses from 223.1.17.128/26 to 223.1.17.191/26.
Subnet 3 has at least 12 interfaces, so 4 bits can support up to 16 addresses. The host addresses range from 192 to 207, which is 11000000 to 11001111. The address range varies by the last 5 bits out of all 32 bits, so the prefix range is $32 - 4 = 28$. Subnet 3 consists of the 16 addresses from 223.1.17.192/28 to 223.1.17.207/28.