COMP 362 Computer Networks II: Quiz II

Nov 6, 2001

Name ……………………………..……. Student id: ………..…………. Email: ……………… Lab: ……

Question 1: (12 Marks, 9 for a. and 3 for b.)

You are hired to design a reliable byte-stream protocol that uses a sliding window (like TCP). This protocol will run over a 100Mbps network. The RTT of the network is 100ms, and the maximum segment lifetime is 60 seconds.

a) How many bits would you include in the Advertised_window and Sequence_number fields of your protocol header?Justify briefly.
Hints: consider keeping the pipe full and avoiding wrap arounds
b) According to you, if the maximum segment lifetime were not an issue what would be the minimum number of bits required for Sequence_number. Justify briefly.

Your answers here:

a) The advertised window should be large enough to keep the pipe full (i.e., should equal the bandwidth delay product),

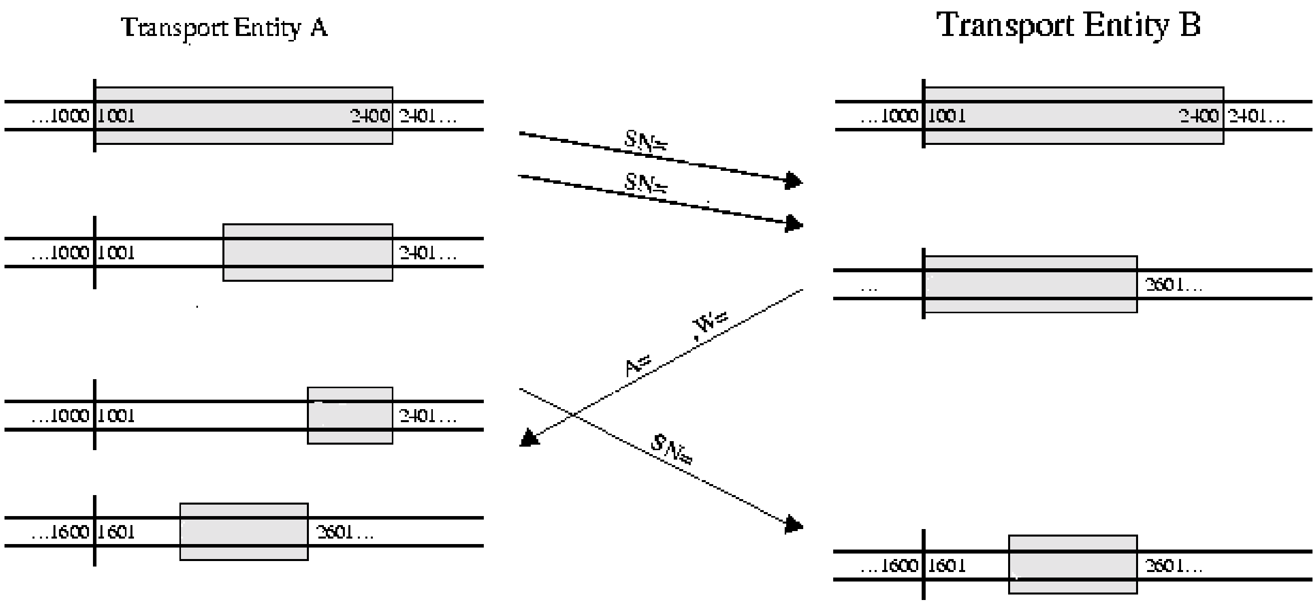
Bandwidth x delay product = 100ms x 100 Mbps = 10Mb = 1.25 MB of data.
This requires 21 bits ($2^{20=}$ 1MB and 2^{21} = 2,097,152) for the advertised window.
The sequence numbers must not wrap around in the maximum segment lifetime.
In 60 seconds, 750 MB can be transmitted. 30 bits allow a sequence space of 1024 MB and so will not wrap in 60 seconds (29 bits allow only 512MB)

b) If the MSL were not an issue the sequence number space should at least be twice the window size to account for outstanding Data and outstanding ACKs.

Example: with 2 bits sequence number, the sender sends packet 0, 1, 2, and 3. The receiver receives them in sequence and without error, and ACKs each one of them and the next packet it expects is numbered 0 (4 mod 2^2). The ACKs are all lost. After a timeout the sender retransmits packets 0 to 3. The receiver accepts them as new packets. This is because the sequnce space is the same as the window size. If we add a single sequence number (which can only be done by doubling the sequence space, i.e. add one more bit to the sequence number field) the problem can be solved. Instead of 3 bits we use 4. Thus after receiving packet number 0 the second time, the receiver knows it is a retransmitted packet because it was expecting packet number 4 (4 mod 2^3)

Question 2: (15 marks 1.5 mark for each of the required numbers plus 3 marks for consistency between the numbers)

Consider the following TCP communication trace where SN stands for the sequence number in the TCP segment header, W the advertised window size and A the ACK number field in the TCP header. The gray boxes represent the window at the sender and at the receiver. Assuming that each segment sent by transport entity A (the sender) contains 300 bytes of data, and that there are no errors or losses, fill the empty spaces (SN, W, A, and the leading edge of the windows) with the right numbers.



Note that because it is not possible to edit the figure below, we adopt the Chinese writing approach -top-down-left-right- in giving the solution.
SN=1001 (first segment of 300 bytes starting at byte number 1001)
SN=1301 (second segment of 300 bytes starting at 1301)
Left edge (next-byte to send) of A's window is at 1601
B Acks the received data (since we assumed no errors) A= 1601, thus window left edge advances to 1601 and advertises a window of W=1000 bytes (starting at byte

1601). How to deduce this? Knowing A= 1601, we can infer from right edge of B's window that the gray box is 1000 bytes wide.

SN=1601, thus the left edge (next-byte to send) of A's window advances to 1901 (since the segments are all 300 bytes long)

Once A receives the Ack from B, the left edge of A's grey area remains at 1901 (but the right edge as we see advanced to 2601)

B's left edge advances to 1901 once it receives the segment with sequence number 1601