**Lab4**

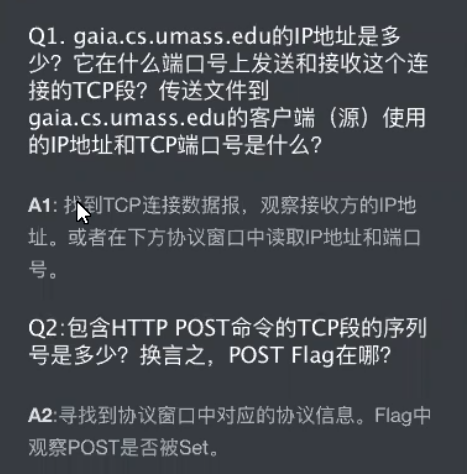
Z5216214

Haojin Guo

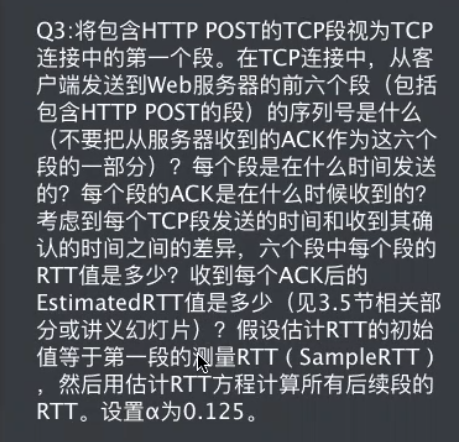
**Exercise 1: Understanding TCP using Wireshark**

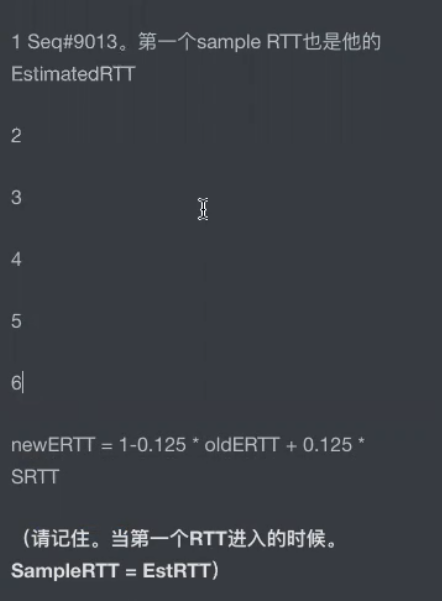
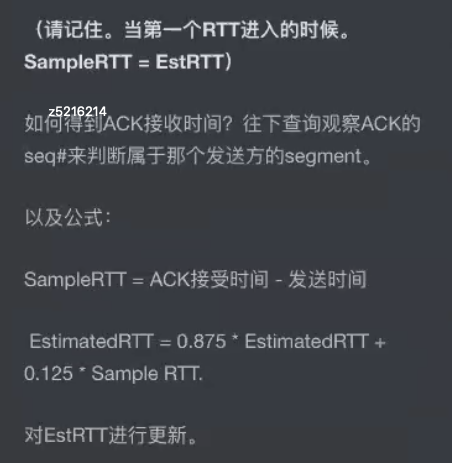
**IMPORTANT NOTE:**Do the sequence numbers for the sender and receiver start from zero? The reason for this is that Wireshark by default scales down all real sequence numbers such that the first segment in the trace file always starts from 0. To turn off this feature, you have to click Edit->Preferences>Protocols->TCP (or Wireshark->Preferences->Protocols->TCP) and then disable the “Relative Sequence Numbers” option. Note that the answers in the solution set will reflect this change. If you conduct the experiment without this change, the sequence numbers that you observe will be different from the ones in the answers. Also, set the time shown in the 2nd column as the "Seconds since the beginning of capture" under view->Time display format.

****

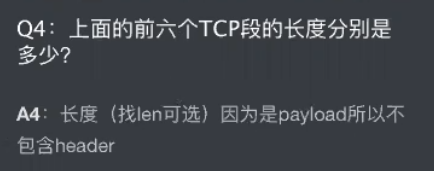
****

*Question 3.*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) sent from the client to the webserver (Do not consider the ACKs received from the server as part of these six segments)? 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 (see relevant parts of Section 3.5 or lecture slides) after the receipt of each ACK? Assume that the initial value of *EstimatedRTT*is equal to the measured RTT ( *SampleRTT*) for the first segment, and then is computed using the *EstimatedRTT*equation for all subsequent segments. Set alpha to 0.125.



*Question 4.*What is the length of each of the first six TCP segments? (same six segments as Q3)

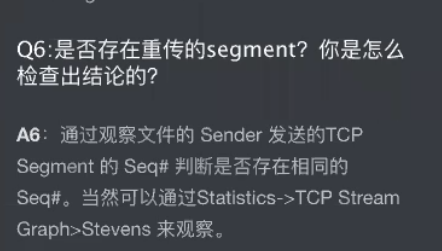


*Question 5.*What is the minimum amount of available buffer space advertised at the receiver for the entire trace? Does the lack of receiver buffer space ever throttle the sender?

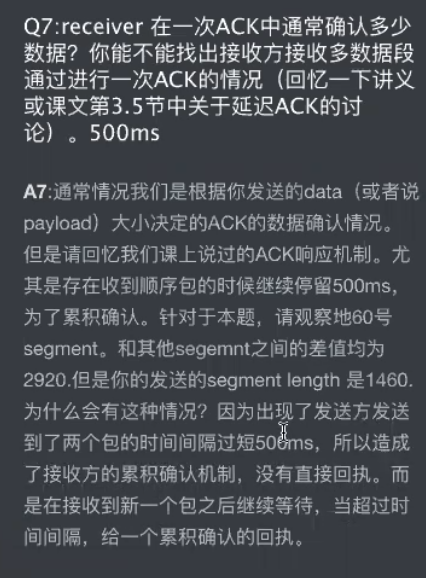




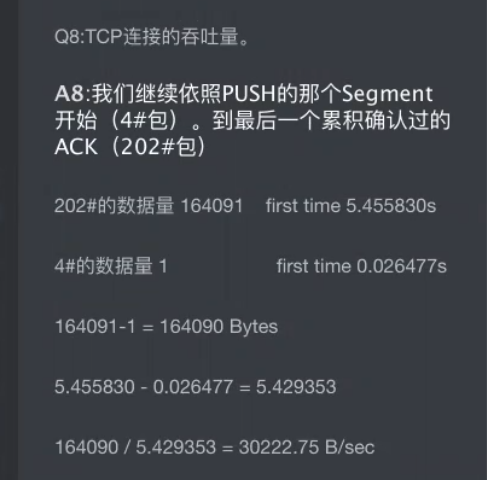
Question 6. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?



*Question 7.*How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (recall the discussion about delayed acks from the lecture notes or Section 3.5 of the text).

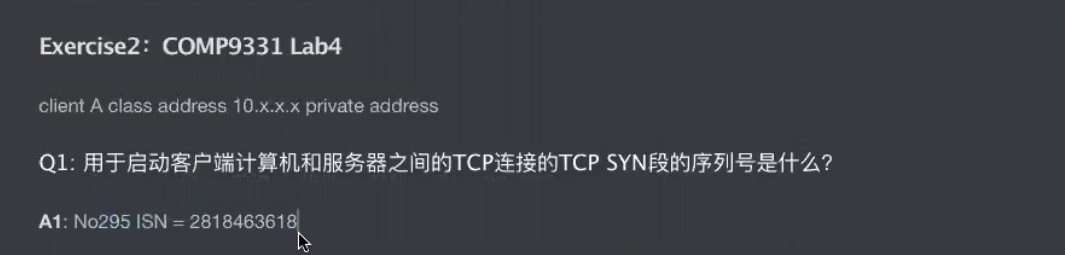


*Question 8.*What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.

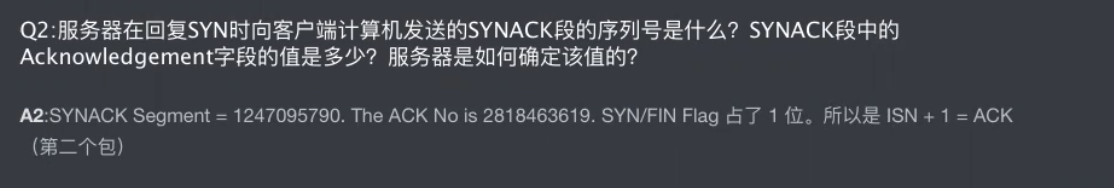


### Exercise 2: TCP Connection Management

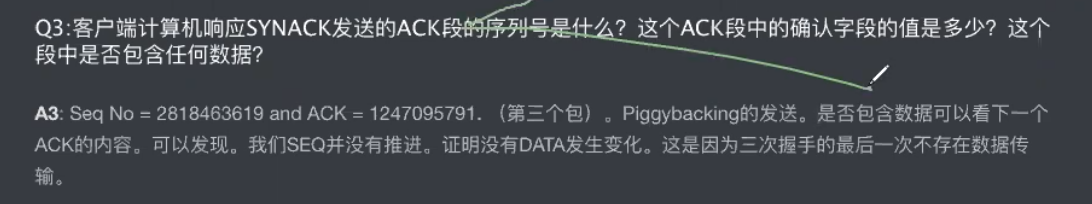
*Question 1*. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and server?



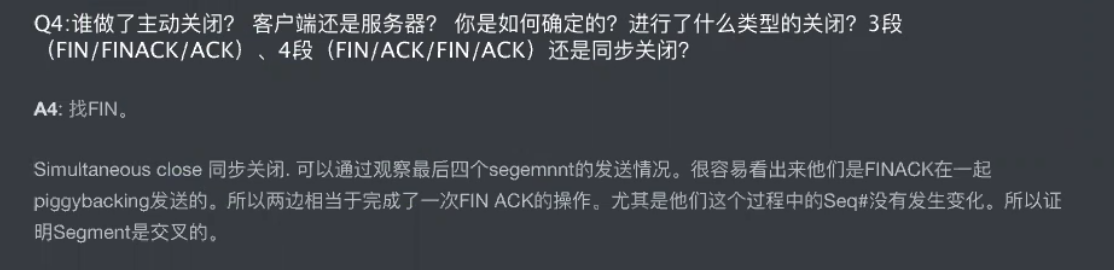
*Question 2.*What is the sequence number of the SYNACK segment sent by the server to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did the server determine that value?

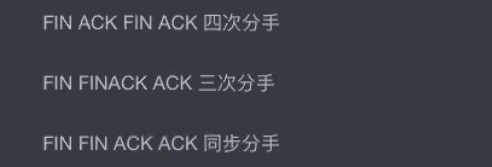


Question 3 . What is the sequence number of the ACK segment sent by the client computer in response to the SYNACK? What is the value of the Acknowledgment field in this ACK segment? Does this segment contain any data?



Question 4 . Who has done the active close? client or the server? how you have determined this? What type of closure has been performed? 3 Segment (FIN/FINACK/ACK), 4 Segment (FIN/ACK/FIN/ACK) or Simultaneous close?





Question 5 . How many data bytes have been transferred from the client to the server and from the server to the client during the whole duration of the connection? What relationship does this have with the Initial Sequence Number and the final ACK received from the other side?

