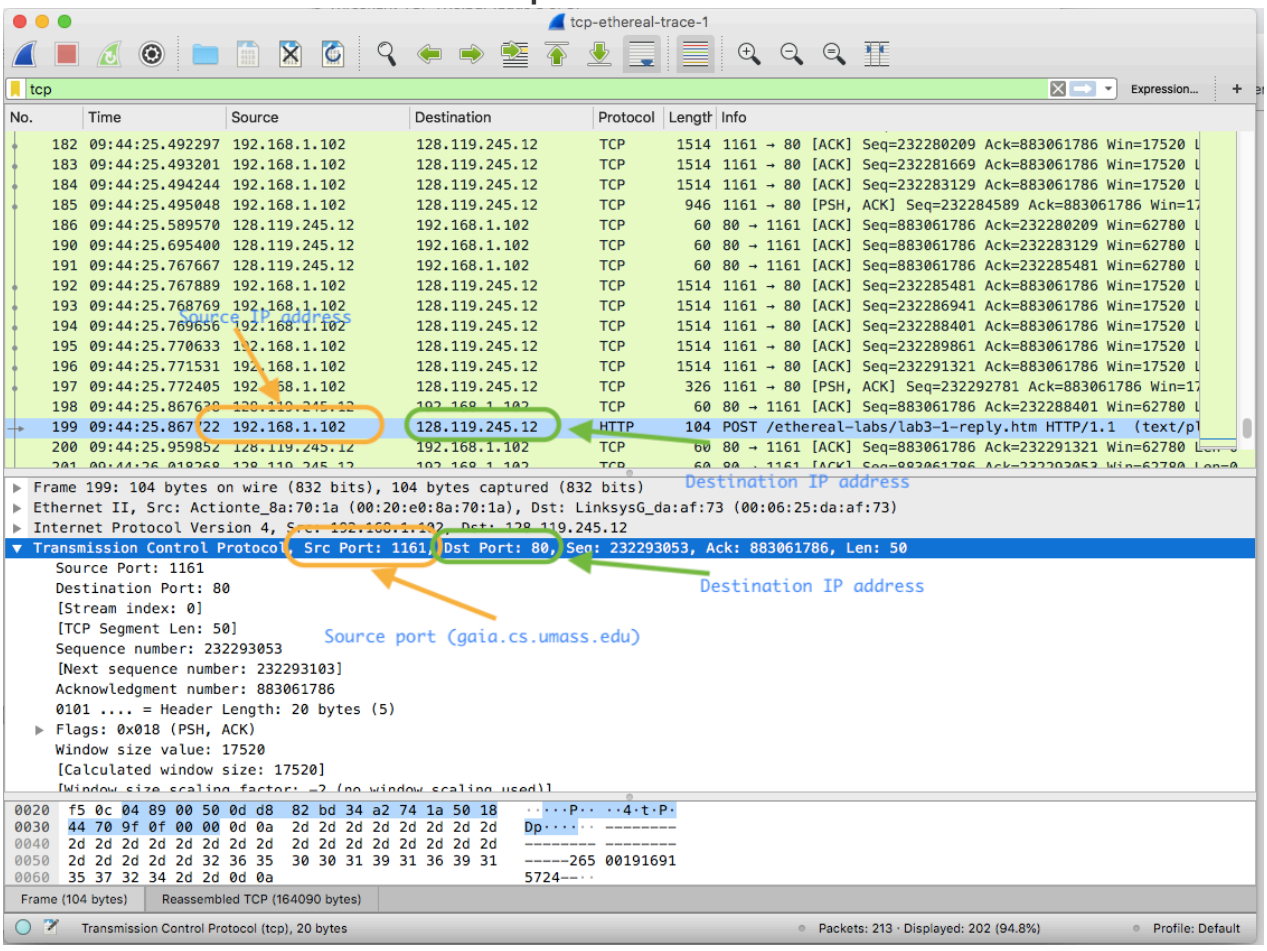
Wireshark Lab : TCP

use tcp-ethereal-trace-1 as reference and set sequence number as absolute seq number

1. What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details of the selected packet header window" (refer to Figure 2 in the "Getting Started with Wireshark" Lab if you're uncertain about the Wireshark windows.

ANS: The IP address 192.168.1.102 TCP port is 1161



2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?

   ANS:

   Sending IP address:192.168.1.102 TCP port: 1161

   Receiving IP address: 128.119.245.12 TCP port: 80

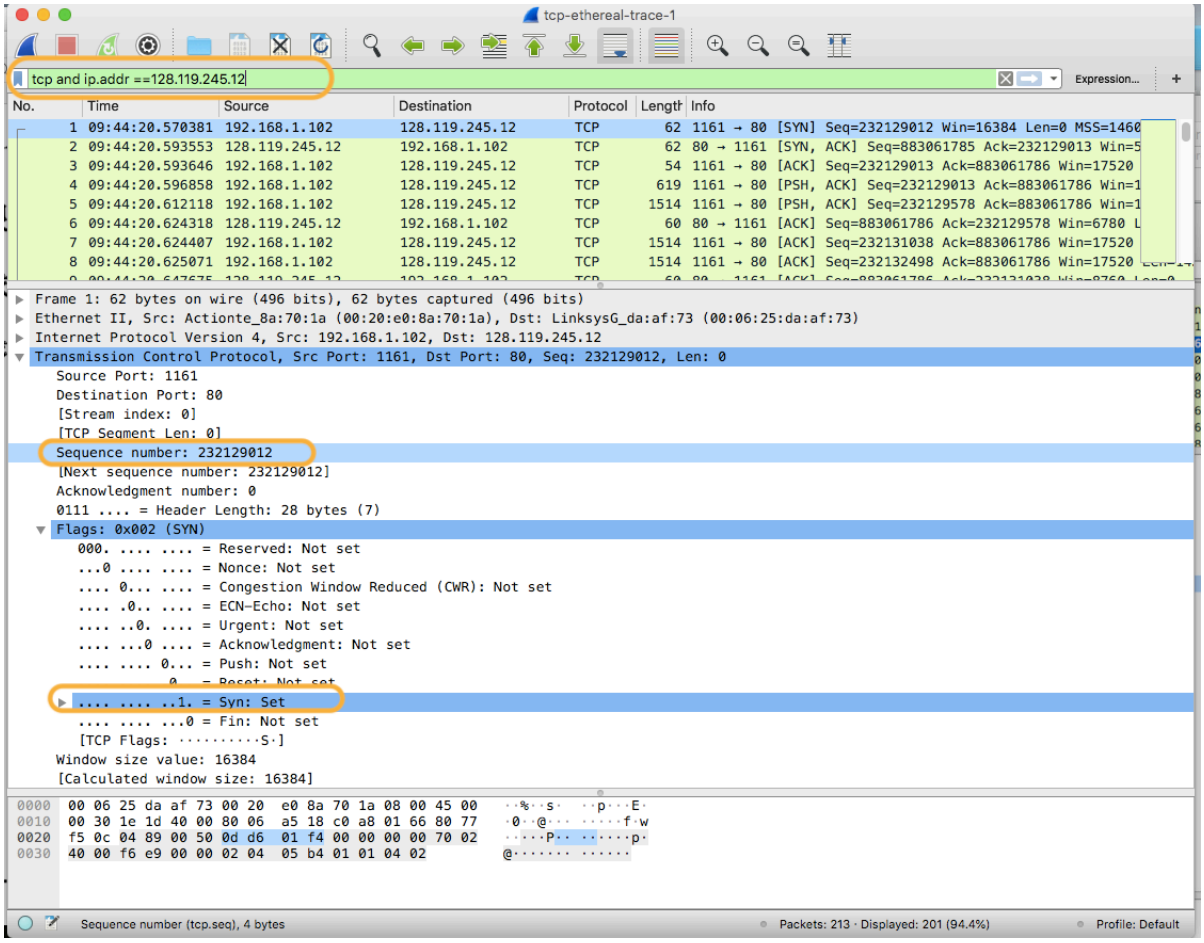
3. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu

   ANS:

4. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?

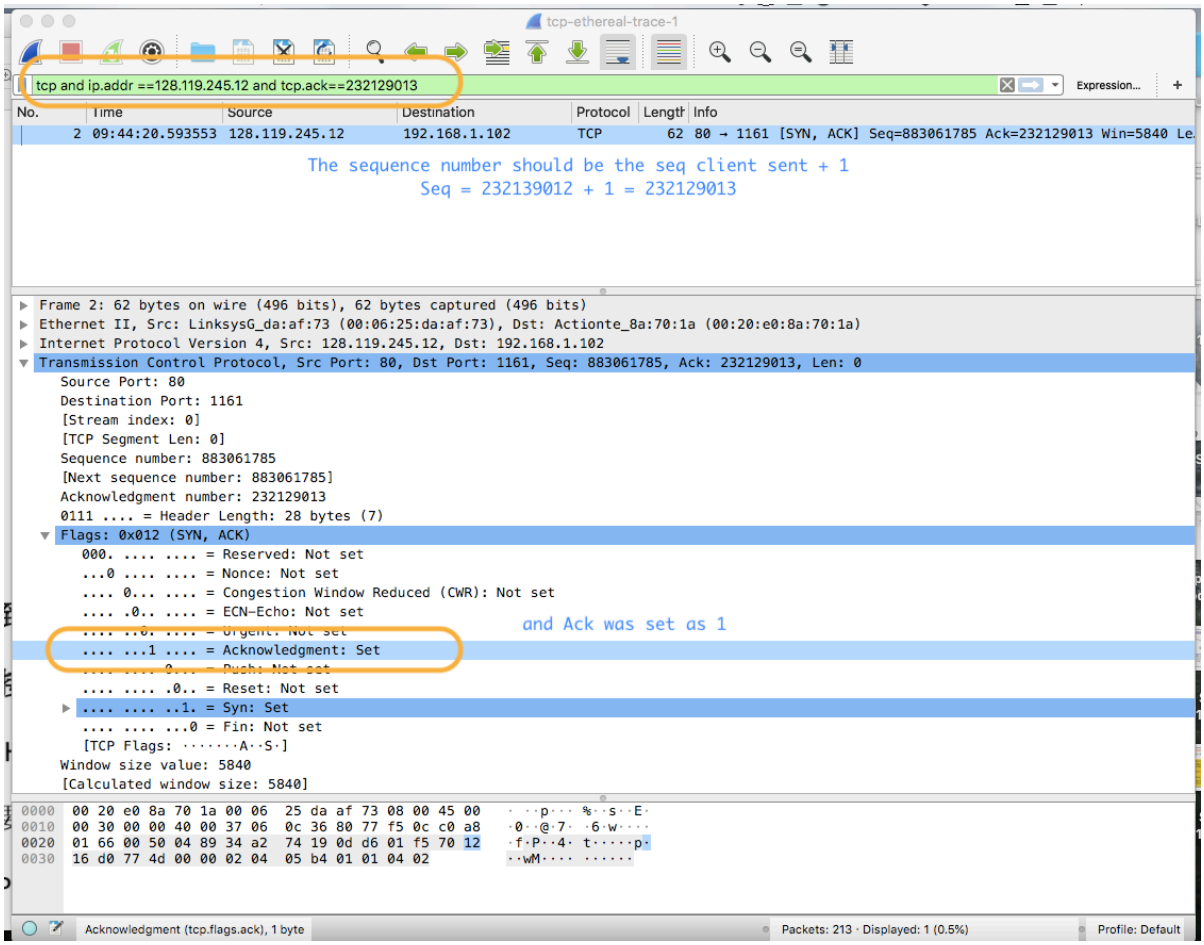
   ANS: Client send SYN to start connection. I searched for the first sent request and find the client set SYN as 1 to establish the connection.

   SEQ = 232129012, which is a random number. This is the first step of three-way handshake.



5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?

ANS:



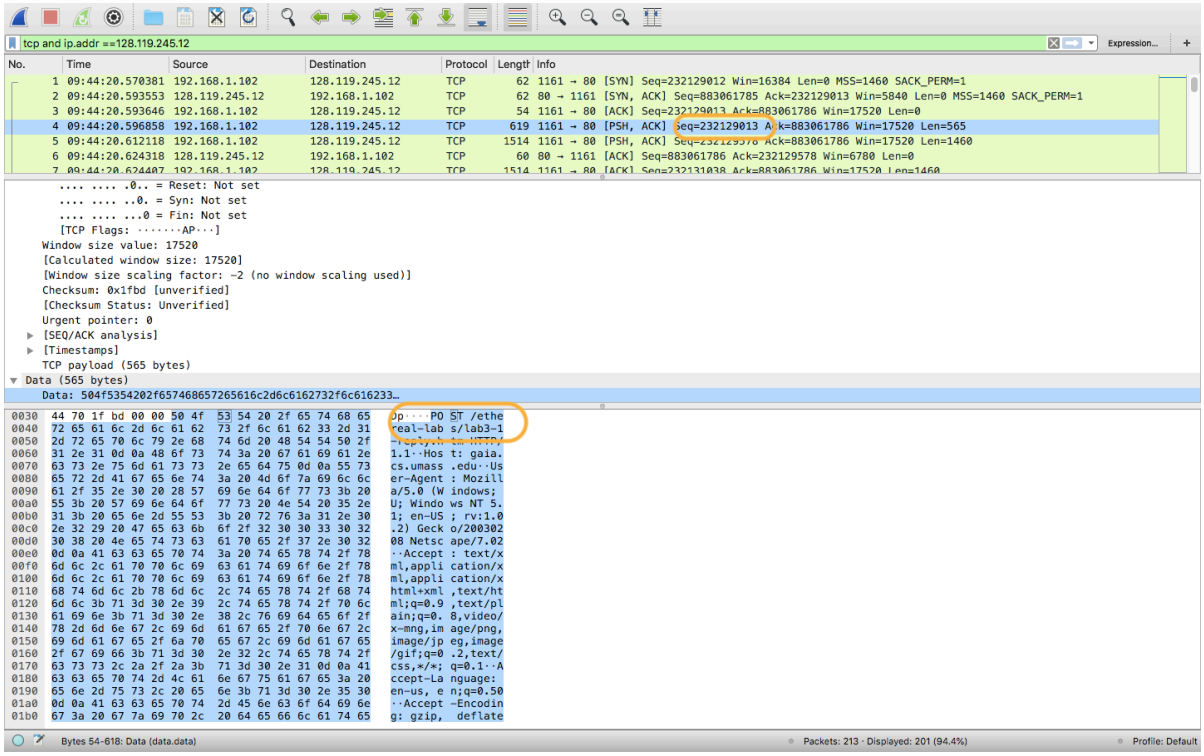
According to the sequence number SEQ = client SEQ + 1 of SYN-ACK response value, the SEQ (sequence number) = 232129013 can be used to search.

And Ack was set as 1, which means the sever has received the client's connection request and sent SYN-ACK to confirm.

This the second step of three-way handshake.

6. What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.

   ANS: The sequence number is 232129013

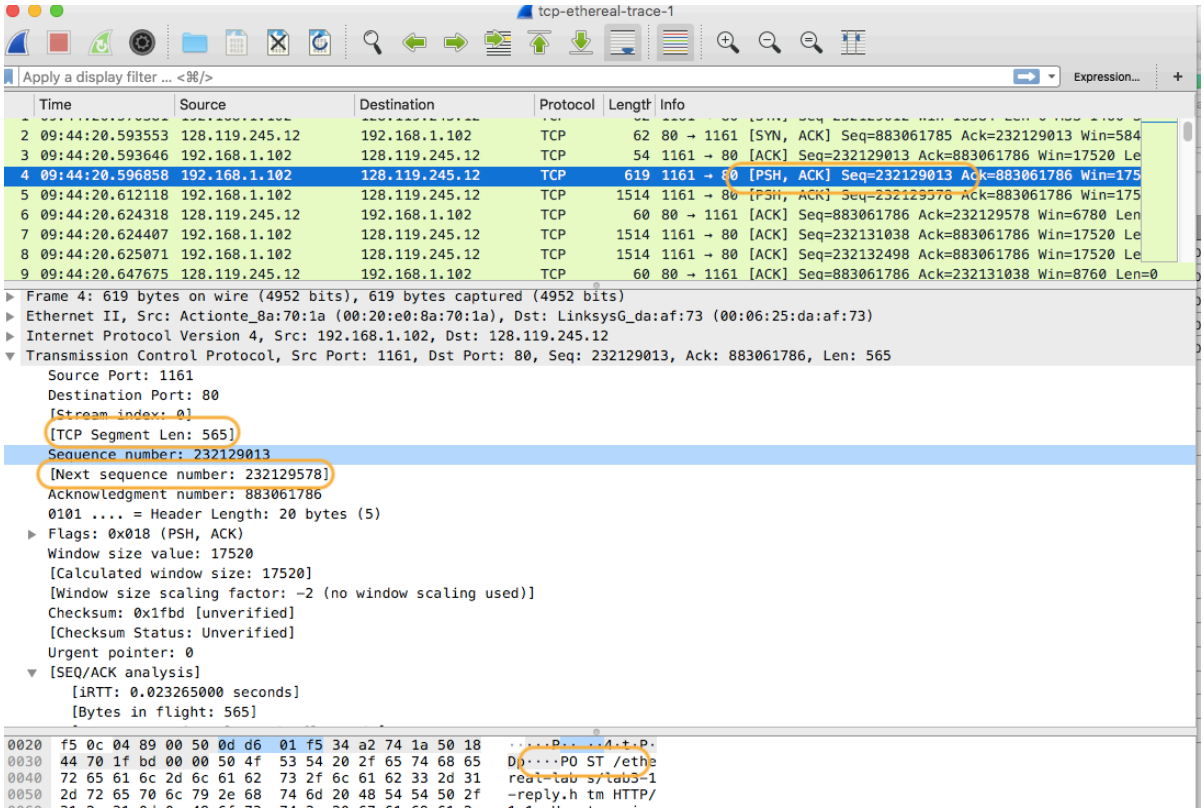


7. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? At what time was each segment sent? When was the ACK for each segment received? 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 Section 3.5.3, page 242 in text) after the receipt of each ACK? Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 242 for all subsequent segments.

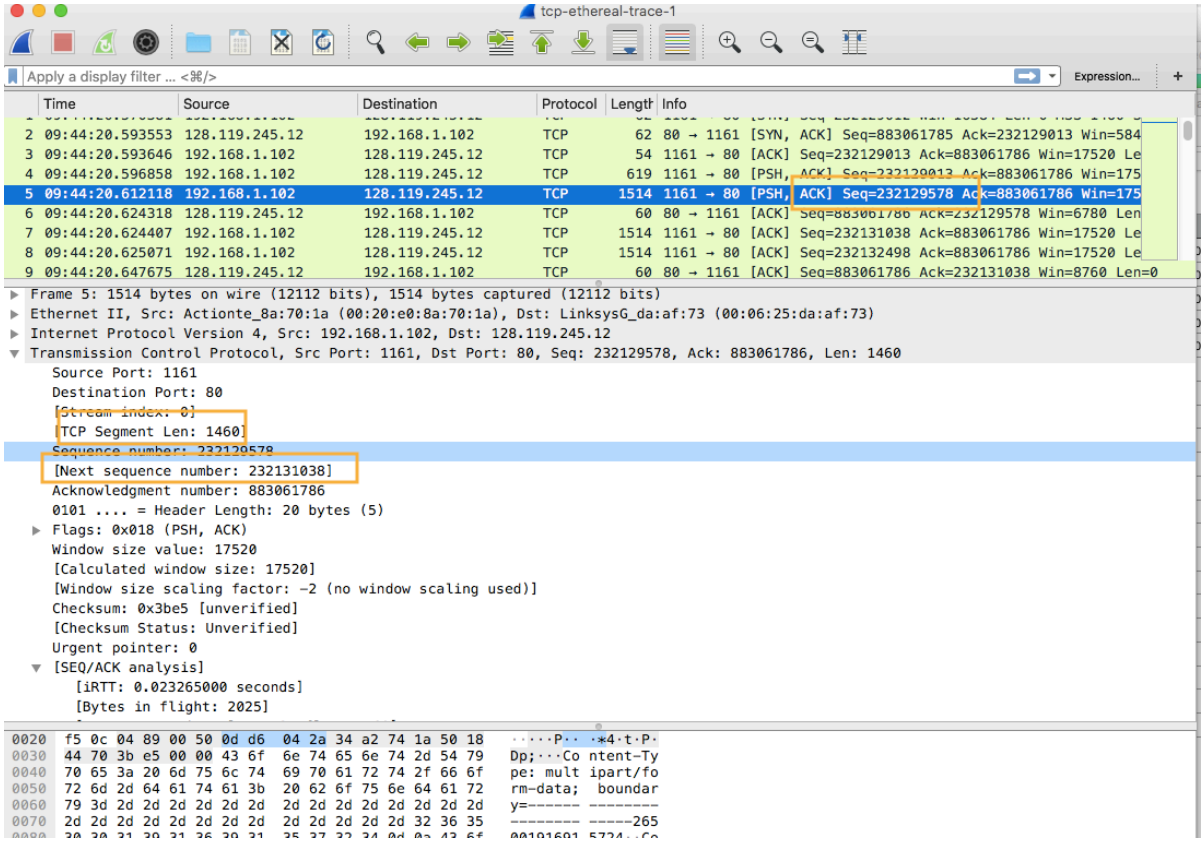
   Note: Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the "listing of captured packets" window that is being sent from the client to the gaia.cs.umass.edu server. Then select: *Statistics->TCP Stream Graph- >Round Trip Time Graph.*

   ANS:

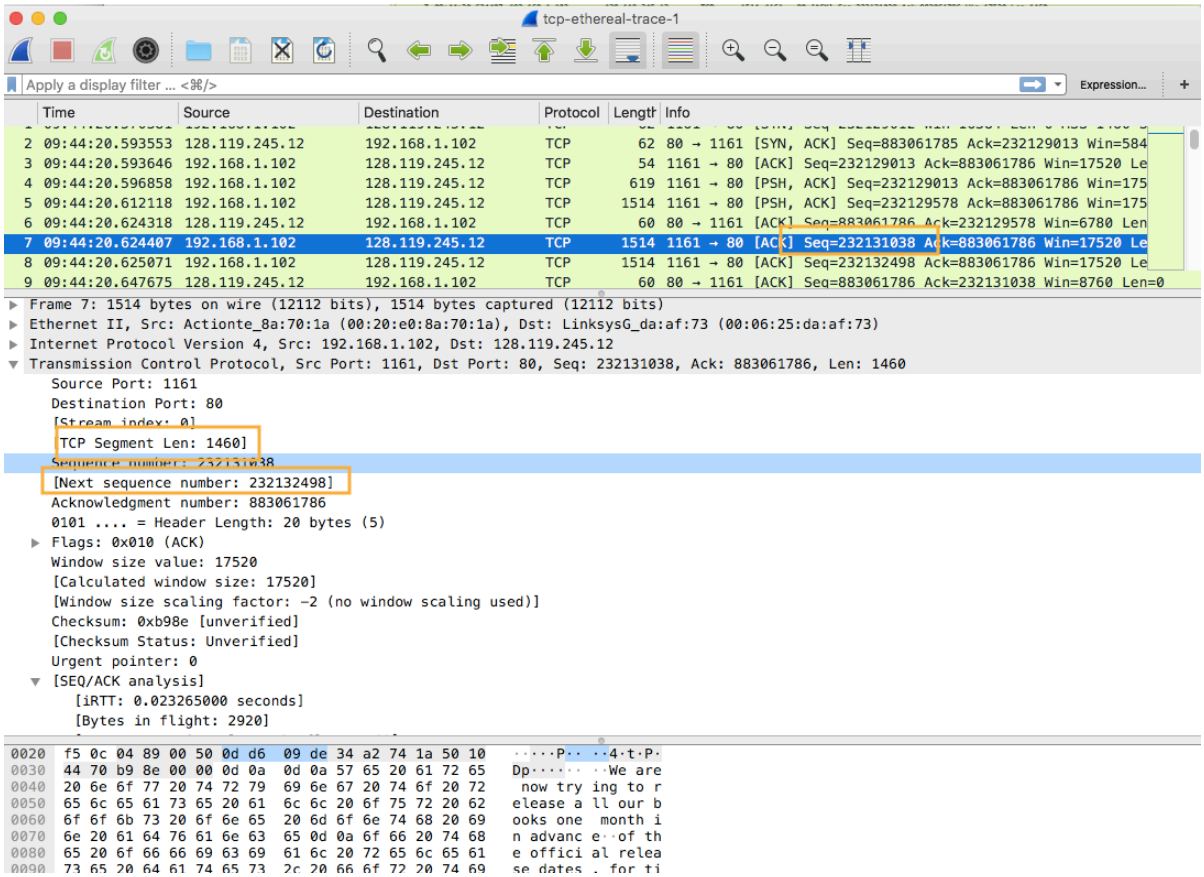
   segment1



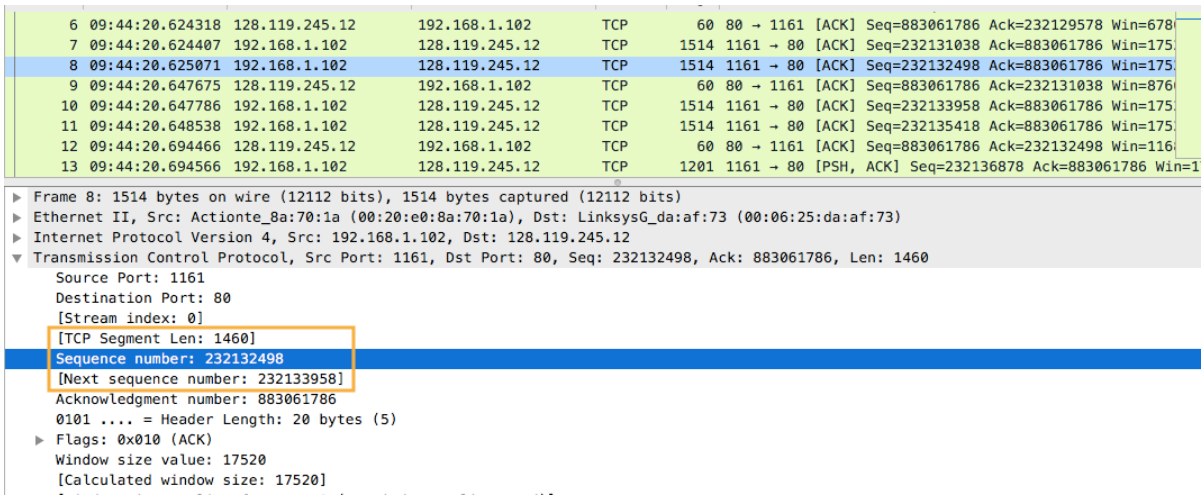
segment2



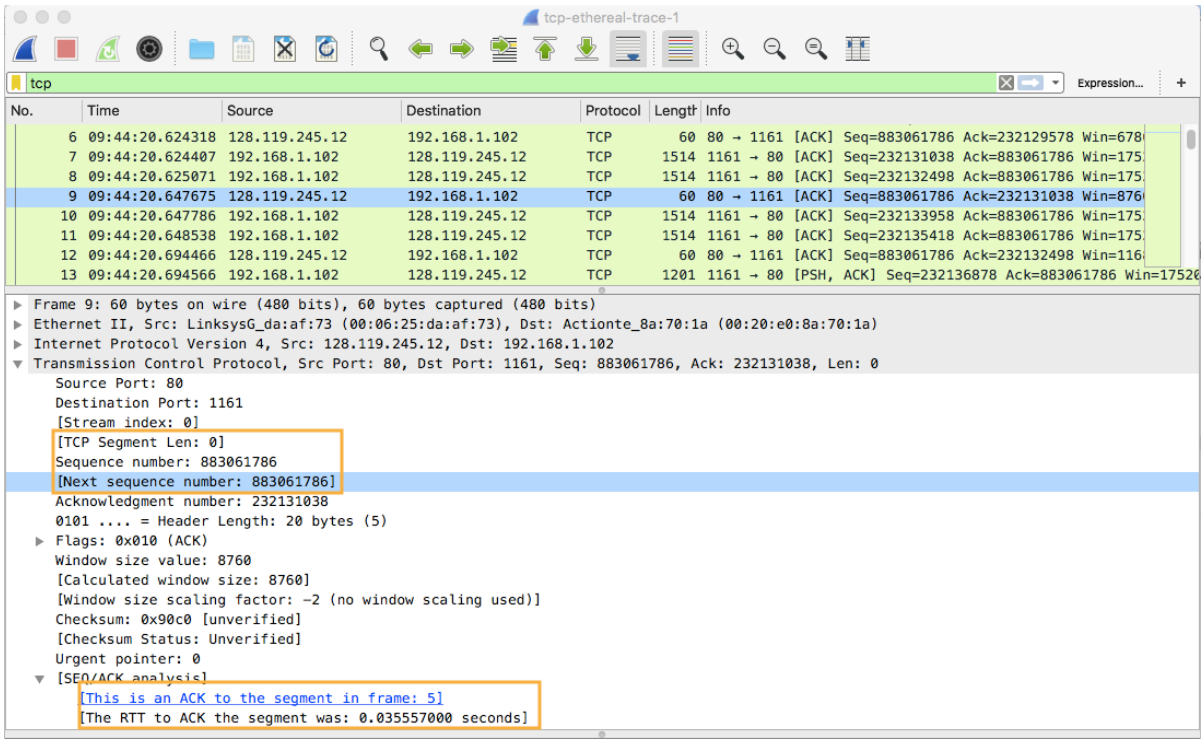
segment3



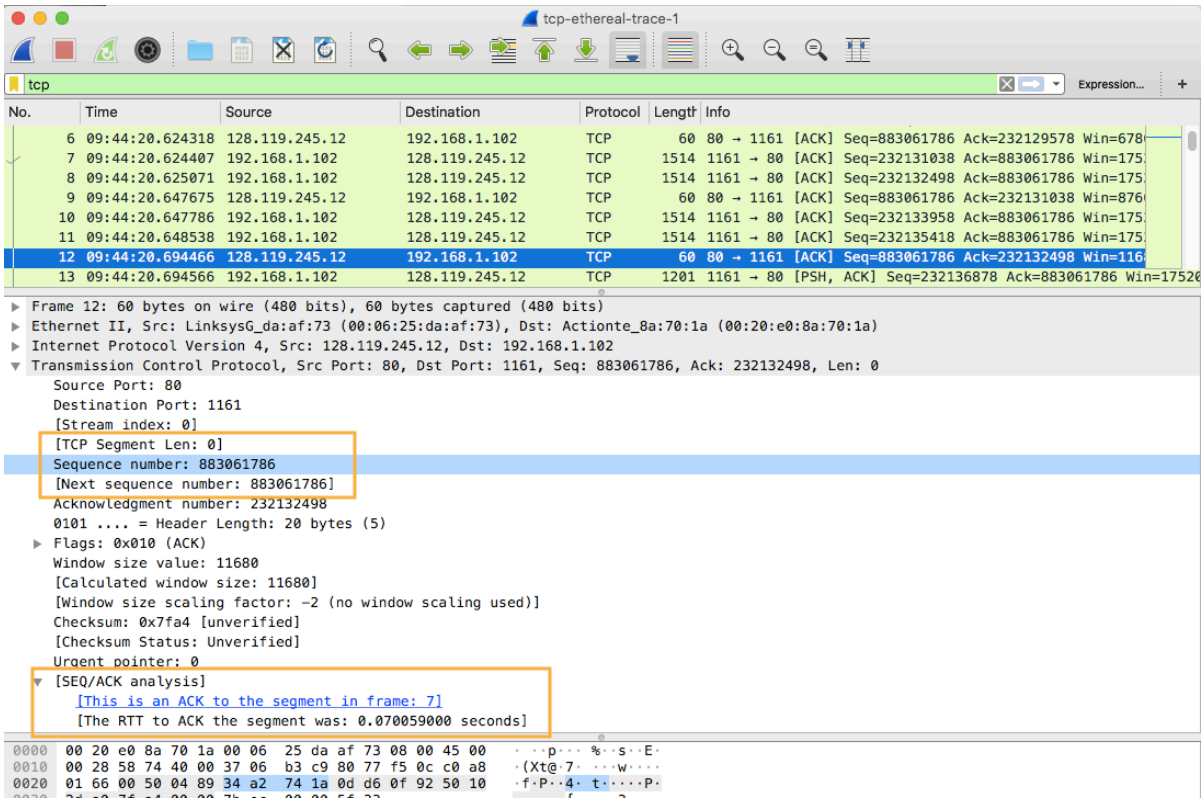
segment4



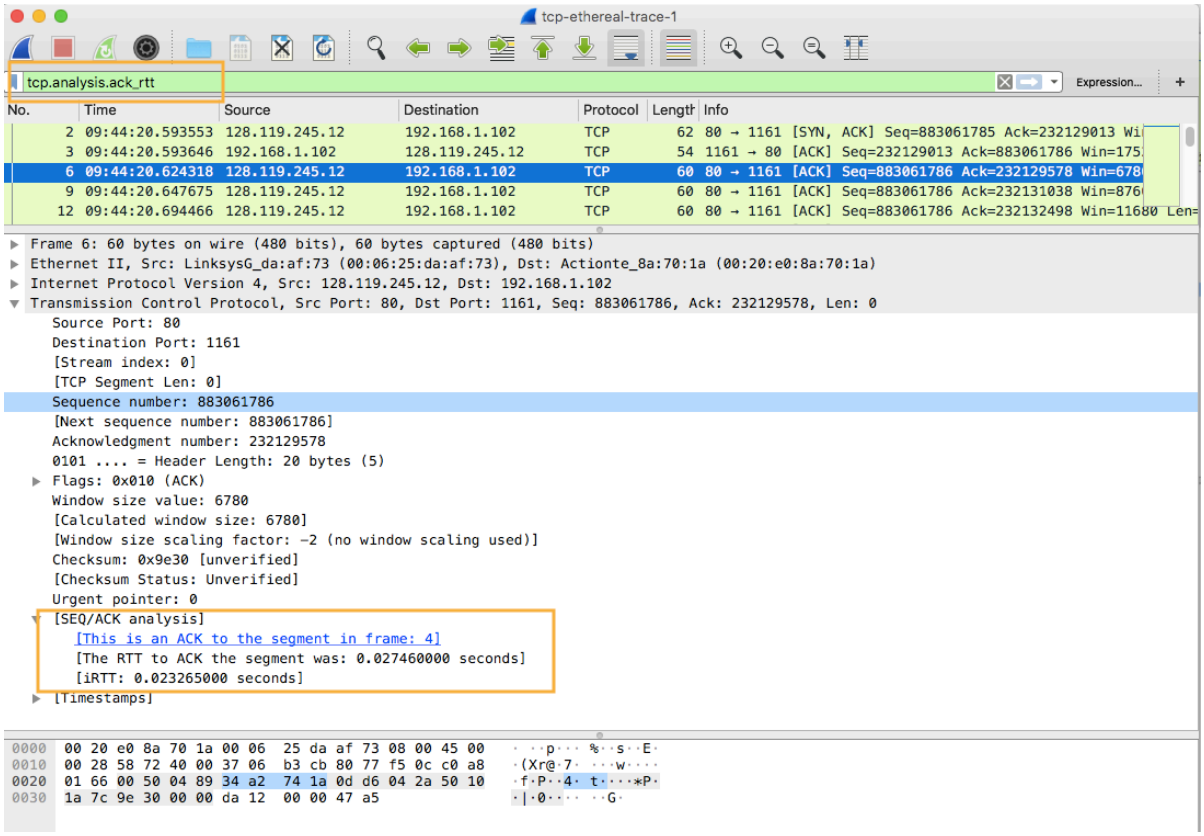
segment5



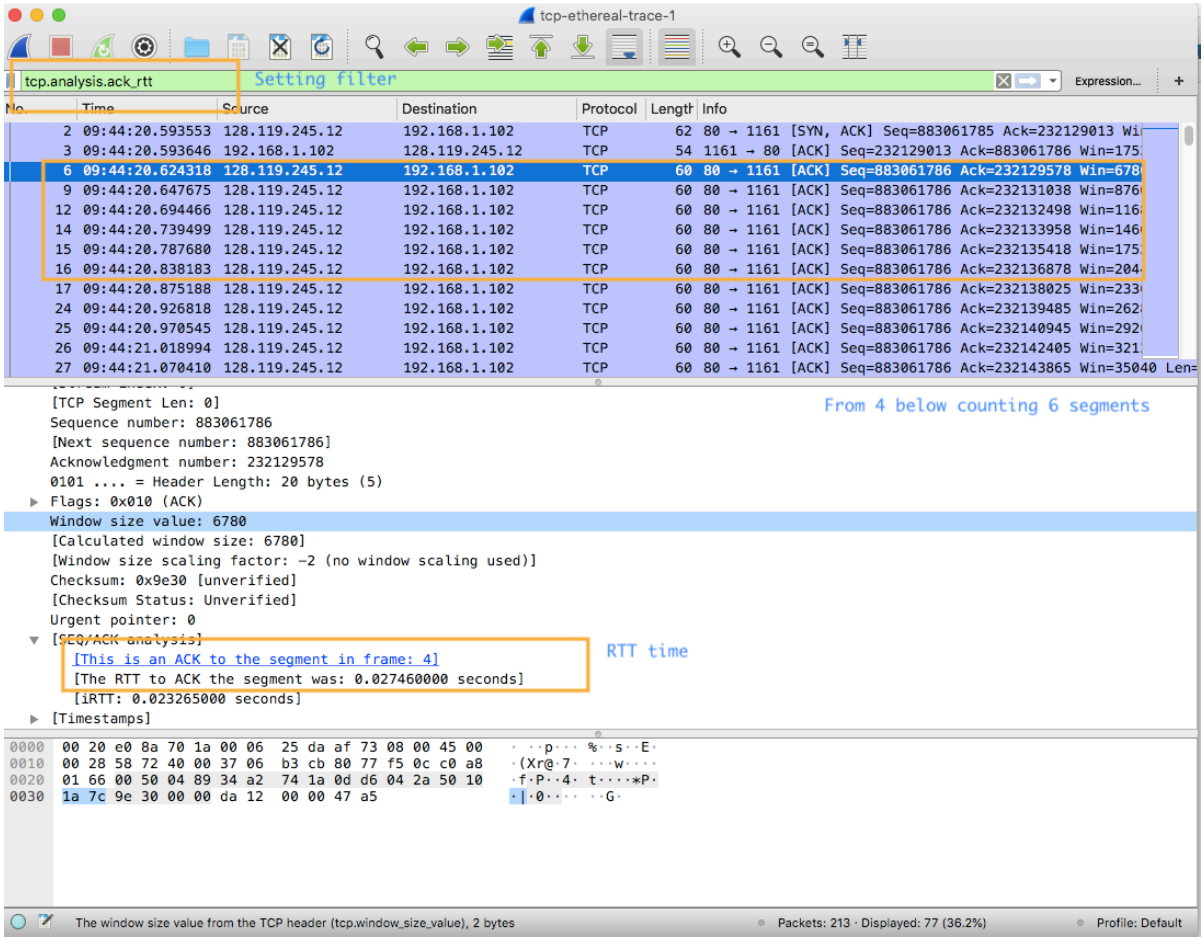
segment6



Then use "tcp.analysis.ack_rtt" to search rtt



Then use "tcp.analysis.ack_rtt" to search rtt



$$EstimatedRTT = \frac{1}{8}RTT + \frac{7}{8}LastEstimatedRTT$$

And you can get the following list

Table 1 Table 1-1

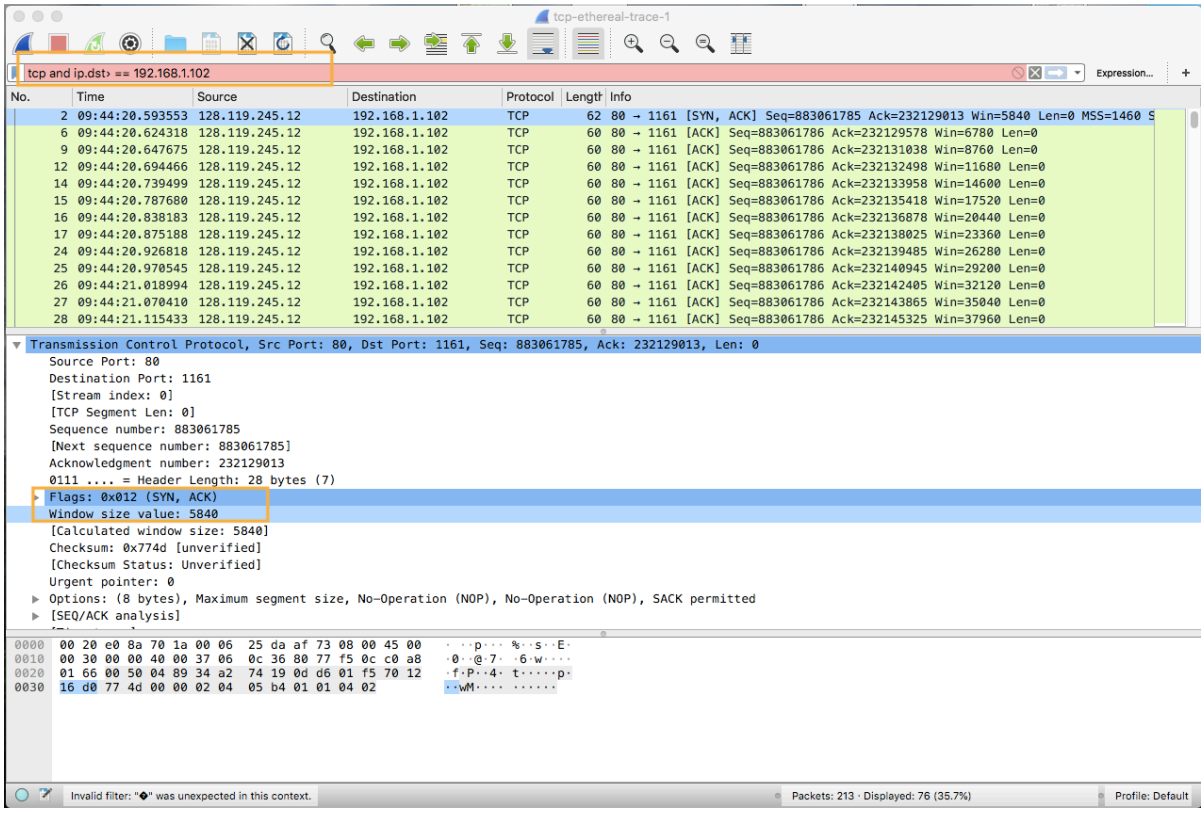
	segment No.	Seq	Len	RTT	EstimatedRTT
1	4	232129013	565	0.027460000	0.02746
2	5	232129578	1460	0.035557000	0.028472125
3	7	232131038	3486	0.070059000	0.033670484
4	8	232132498	1460	0.114428000	0.043765174
5	10	232133958	1460	0.139894000	0.055781277
6	11	232135418	1460	0.189645000	0.072514242

8. What is the length of each of the first six TCP segments?

   ANS see above

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

   ANS：



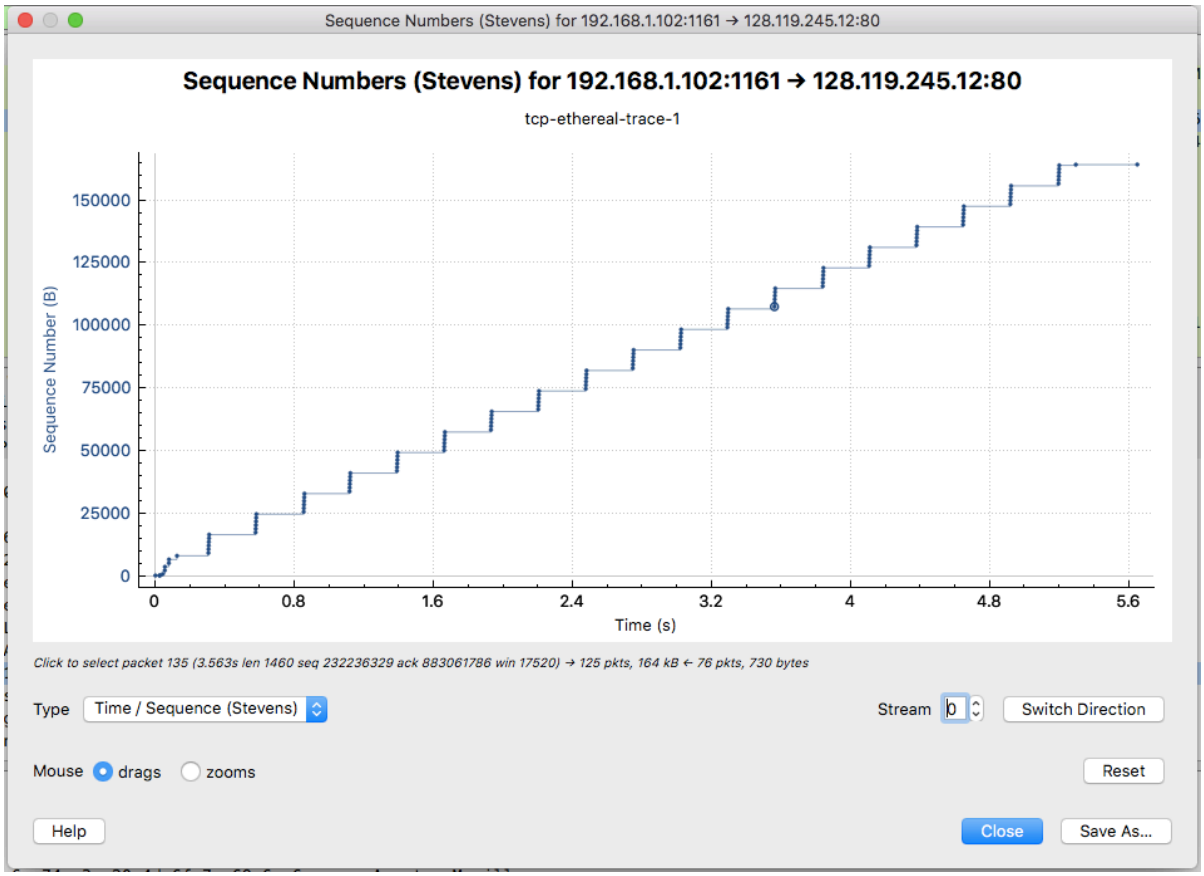
   The receiver window size grows steadily til a maximum sindow size comes. No throttle is made due to the lack of buffer space.

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

    ANS: no

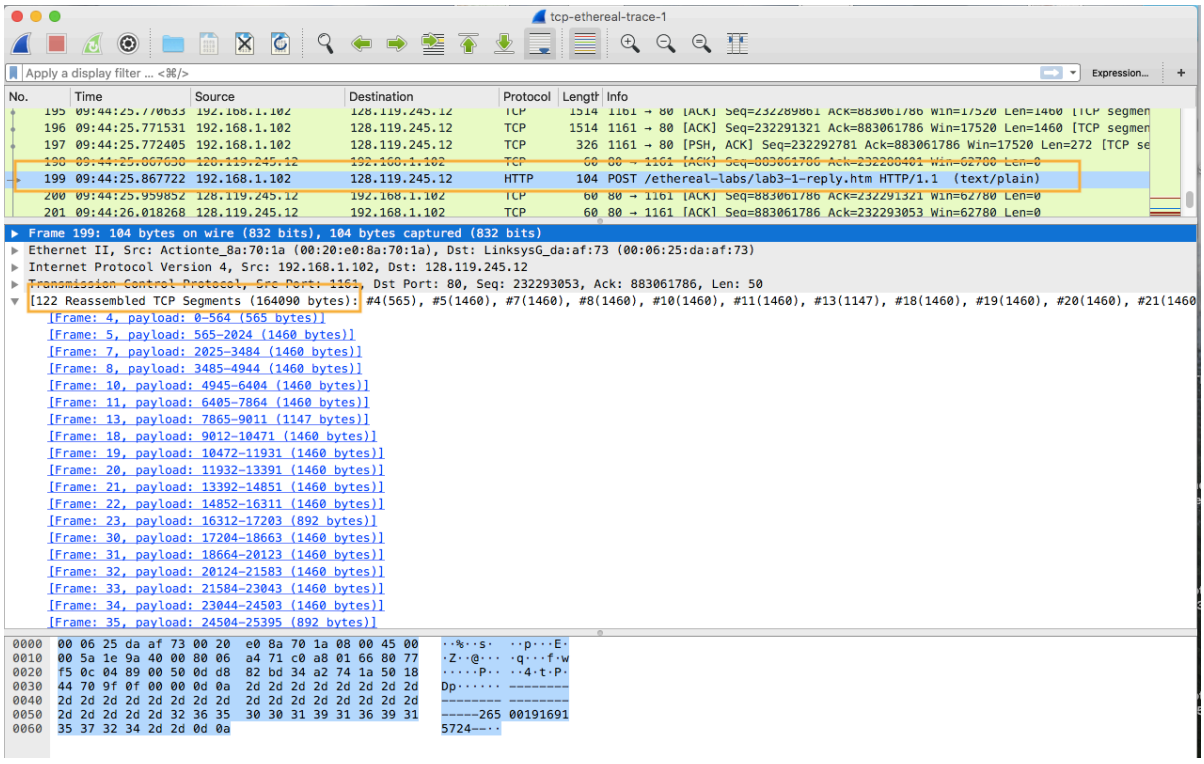
    menu bar -> Statistic -> TCP -> Stevens

    Because the sequence number is steadly increasing.



11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 250 in the text).

    ANS: 122 reassembled TCP segment, totally 164090 bytes.



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

ANS: $Throughput = \frac{AmountOfDataTransmitted}{TimeIncurred}$

$Throughput = frac1640905.297341000 = 30975.9179bytes/s = 30.976kb/s$

13. Use the *Time-Sequence-Graph(Stevens*) plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Can you identify where TCP's slowstart phase begins and ends, and where congestion avoidance takes over? Comment on ways in which the measured data differs from the idealized behavior of TCP that we've studied in the text.

    ANS:

14. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu

    ANS: