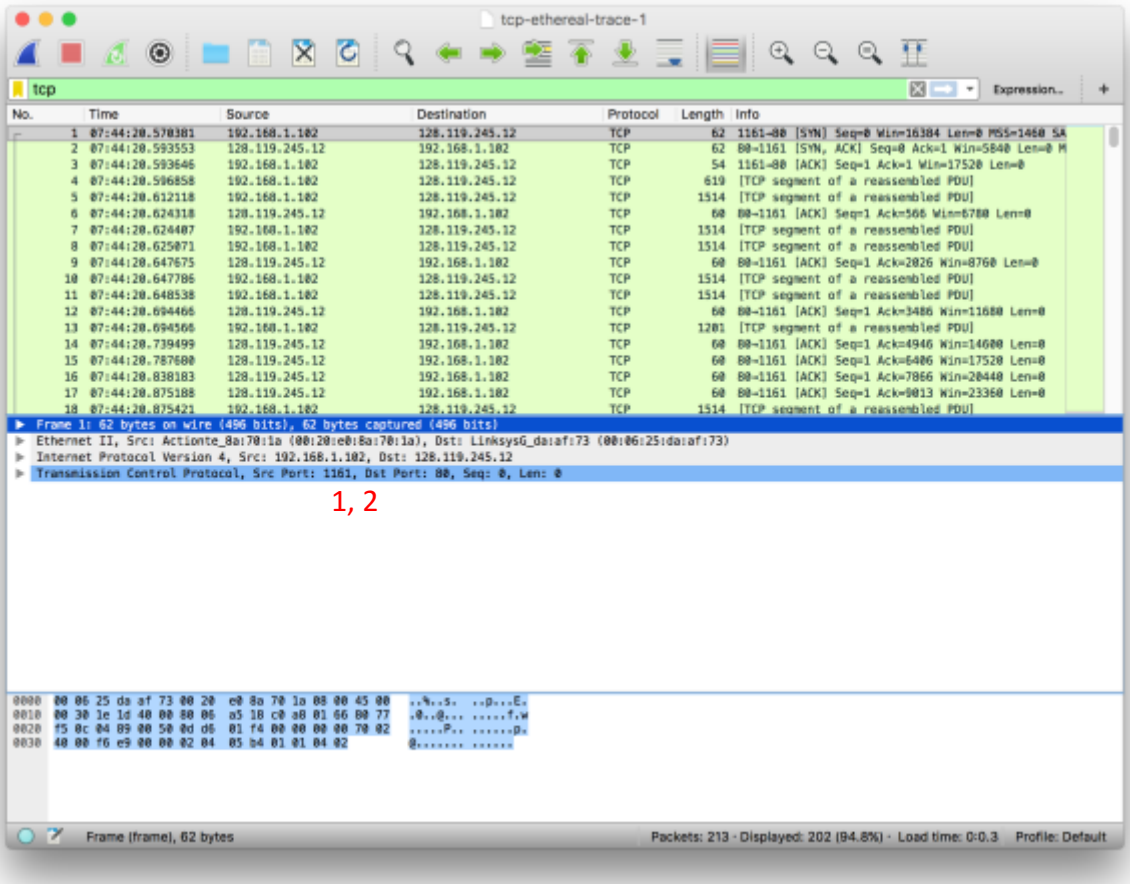
Rob Navarro
CS 372

LAB 3

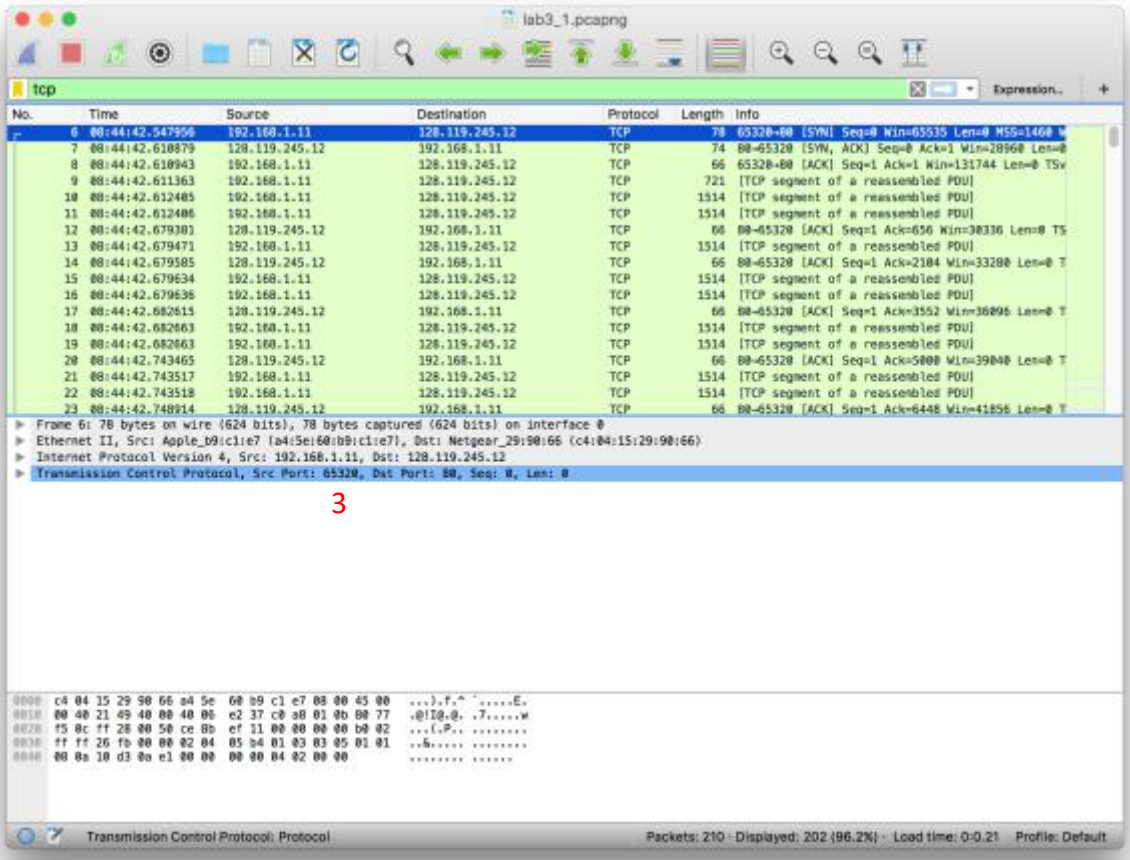
1. What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details 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.
   Answer: The client IP address is 192.168.1.102 and the 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?
   Answer: The IP address of gaia.cs.umass.edu is 128.119.245.12 and the TCP port is 80.



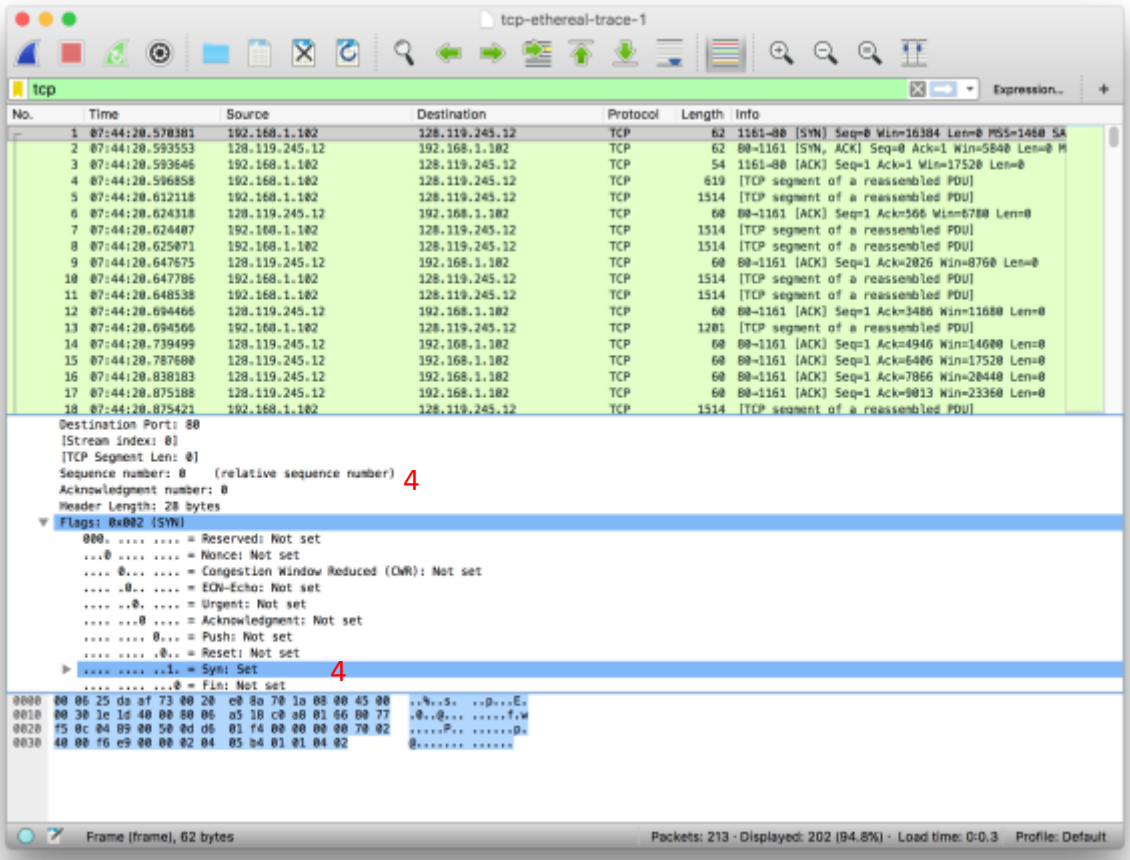
3. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?
   Answer: The IP of my client is 192.168.1.11 and the TCP port is 65320.

Rob Navarro
CS 372



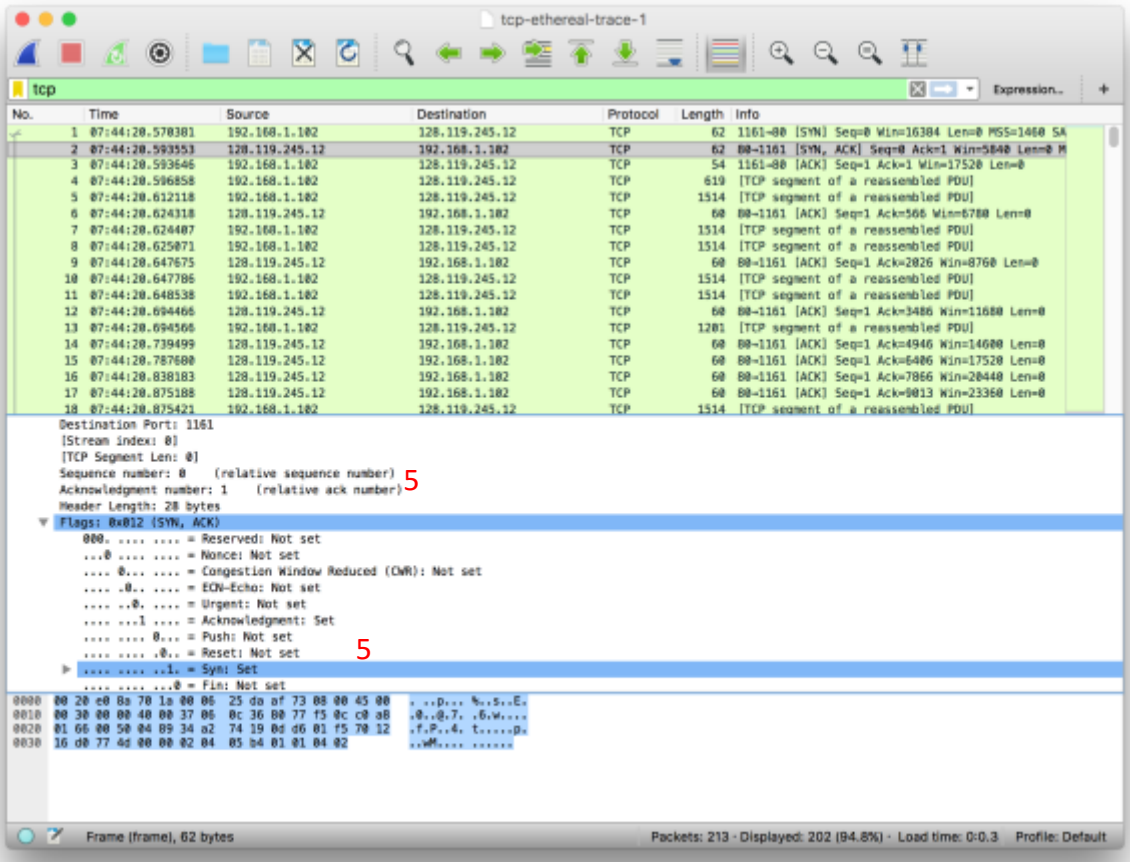
4. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?
   Answer: The sequence number is 0 for this segment. It is identified by the SYN flag being set to 1.

Rob Navarro
CS 372



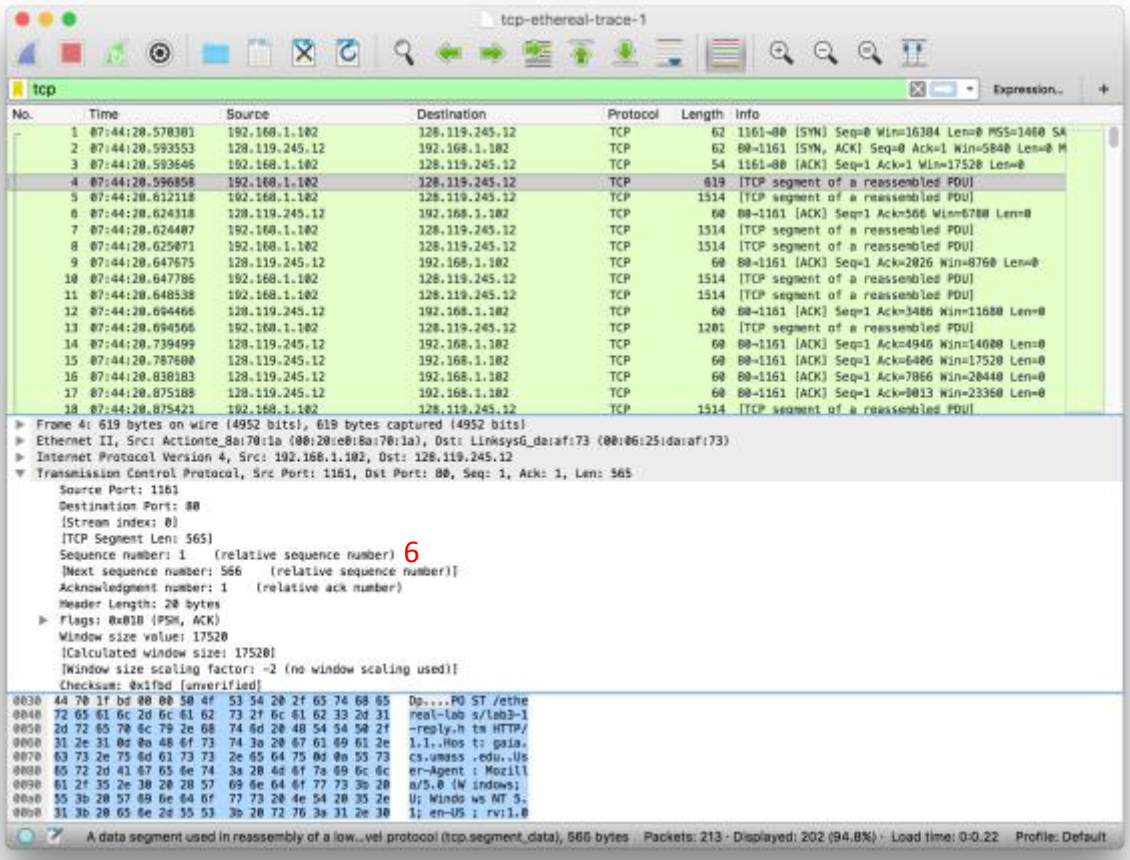
5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?
   Answer: The sequence number is 1 for this segment. The value of the acknowledgment field is 1. This number was determined by gaia.cs.umass.edu by adding 1 to the sequence number. The sequence is identified as SYNACK because both the SYN and Acknowledgement flags are set to 1.

Rob Navarro
CS 372



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.
   Answer: The sequence number is 1.

Rob Navarro
CS 372



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.
   Answer: If the segment containing the HTTP POST is considered to be the first segment, the following table displays the data asked for above:

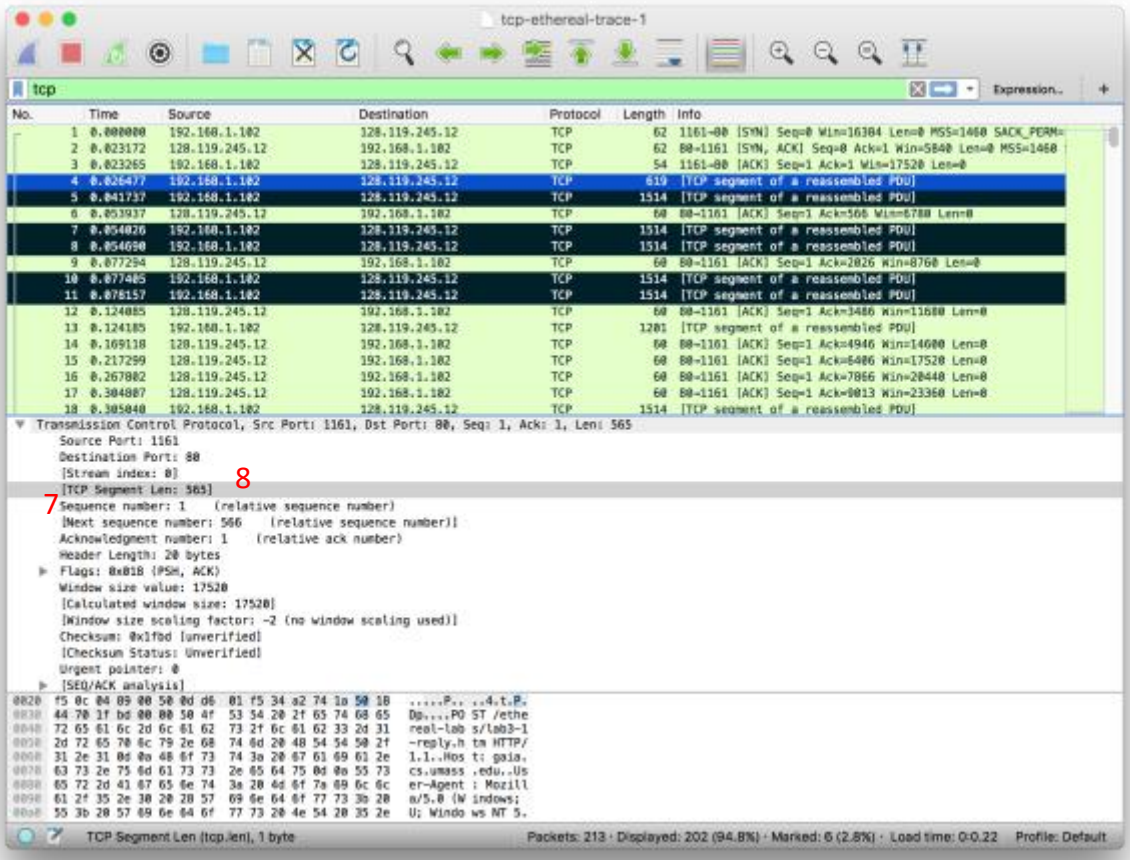
Rob Navarro
CS 372

	Sequence Number	Time Sent	Ack Received	RTT
Segment 1	1	.026477	.053937	0.02746
Segment 2	566	.041737	.077294	0.035557
Segment 3	2026	.054026	.124085	0.070059
Segment 4	3486	.054690	.169118	0.114428
Segment 5	4946	.077405	.217299	0.139894
Segment 6	6406	.078157	.267802	0.189645

EstimatedRTT = 0.875 * EstimatedRTT + 0.125 * sampleRTT
Initial EstimatedRTT = .02746

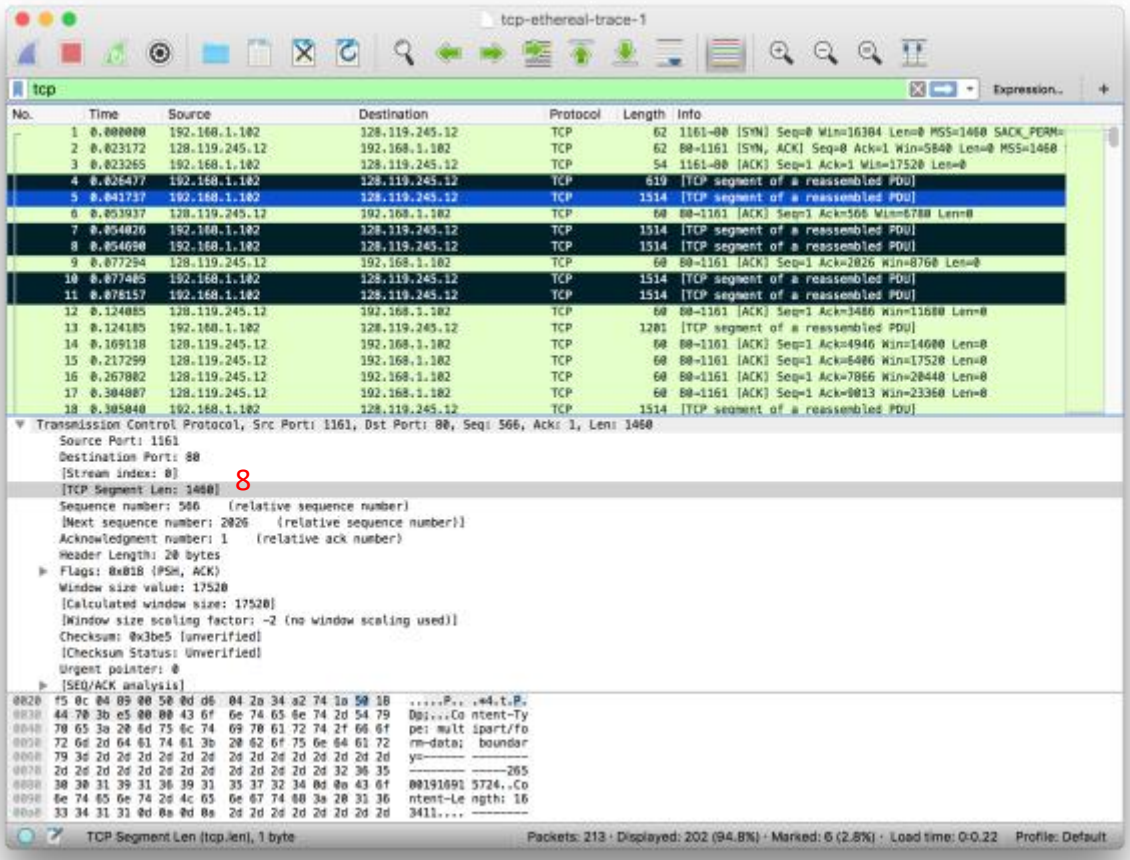
After	EstimatedRTT (seconds)
Segment 1	.02746
Segment 2	.02847
Segment 3	.03367
Segment 4	.04376
Segment 5	.05578
Segment 6	.07251

Rob Navarro
CS 372



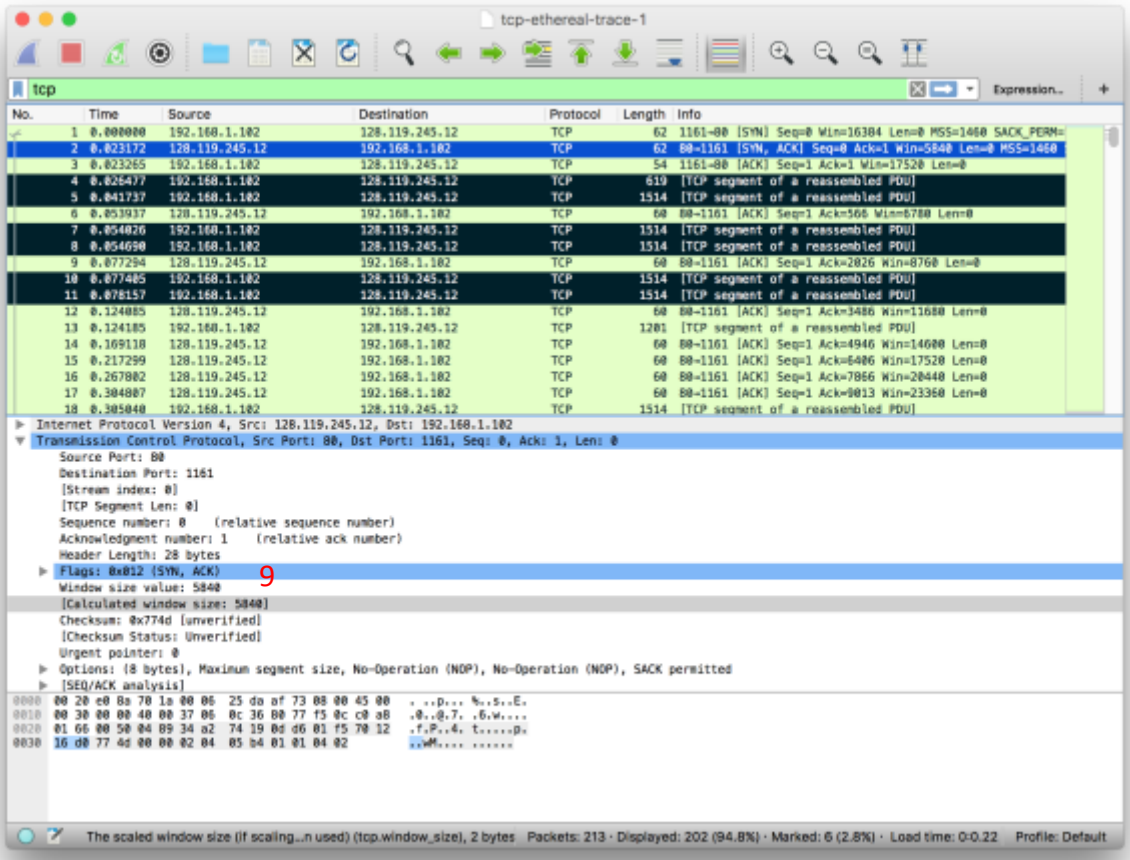
8. What is the length of each of the first six TCP segments?
   Answer: The length of the first segment is 565 bytes (see screenshot above) and each segment after that is 1460 bytes.

Rob Navarro
CS 372



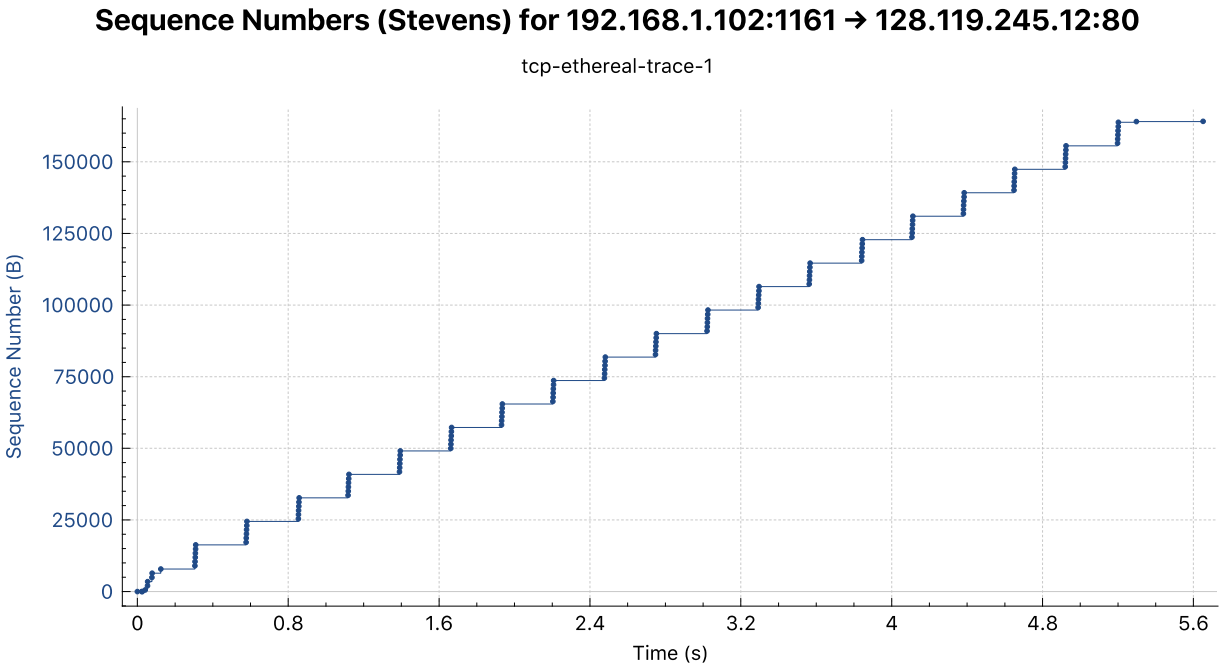
9. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?
   Answer: The minimum amount of buffer space is 5840 bytes, which can be seen in the initial ACK from the server. After reviewing the trace, I did not notice any evidence of the receiver ever throttling the sender. The window size grows throughout the entire transmission process.

Rob Navarro
CS 372



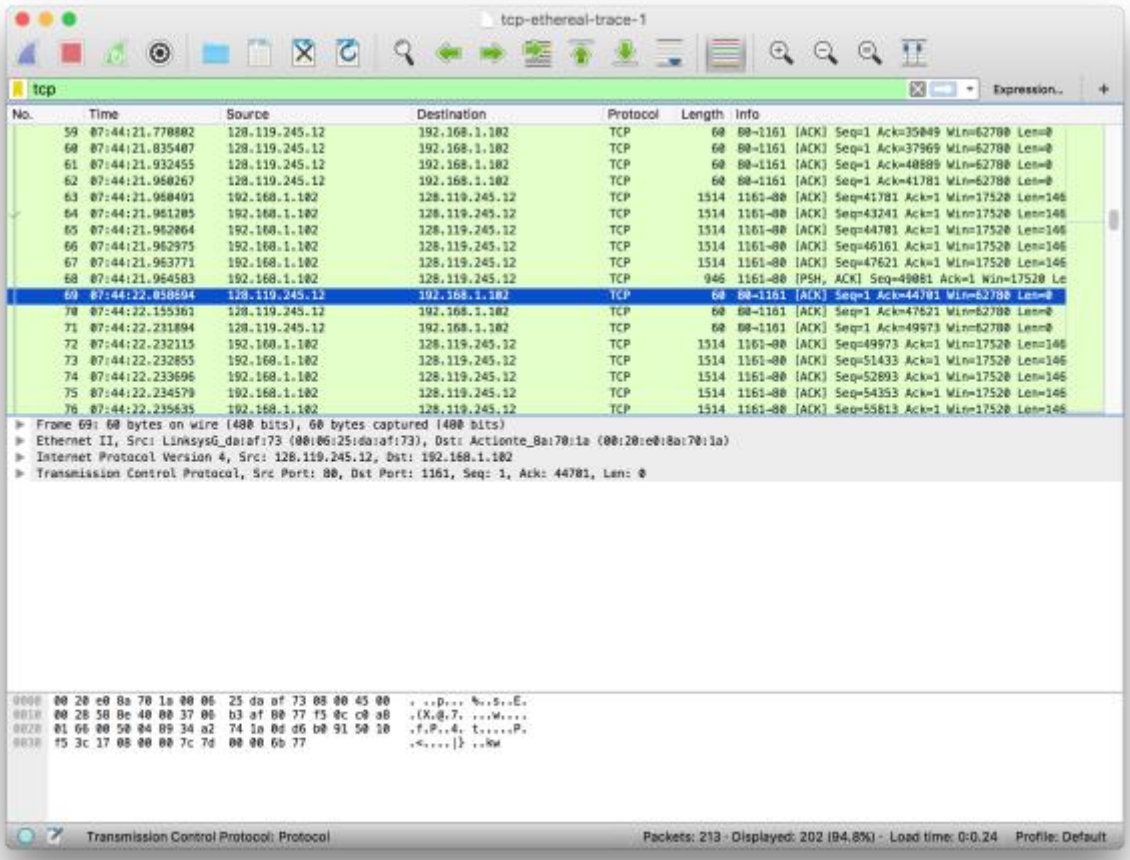
10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?
    Answer: I did not notice any retransmitted segments in the trace file. I confirmed this by reviewing all of the sequence numbers for all the TCP segments. I also reviewed the time-sequence-graph.

Rob Navarro
CS 372



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).
    Answer: Initially the receiver is acknowledging data the size of 1460 bytes. As time goes on though I do see data being acknowledge every other segment. For example, segment 69 acknowledge 2920 bytes.

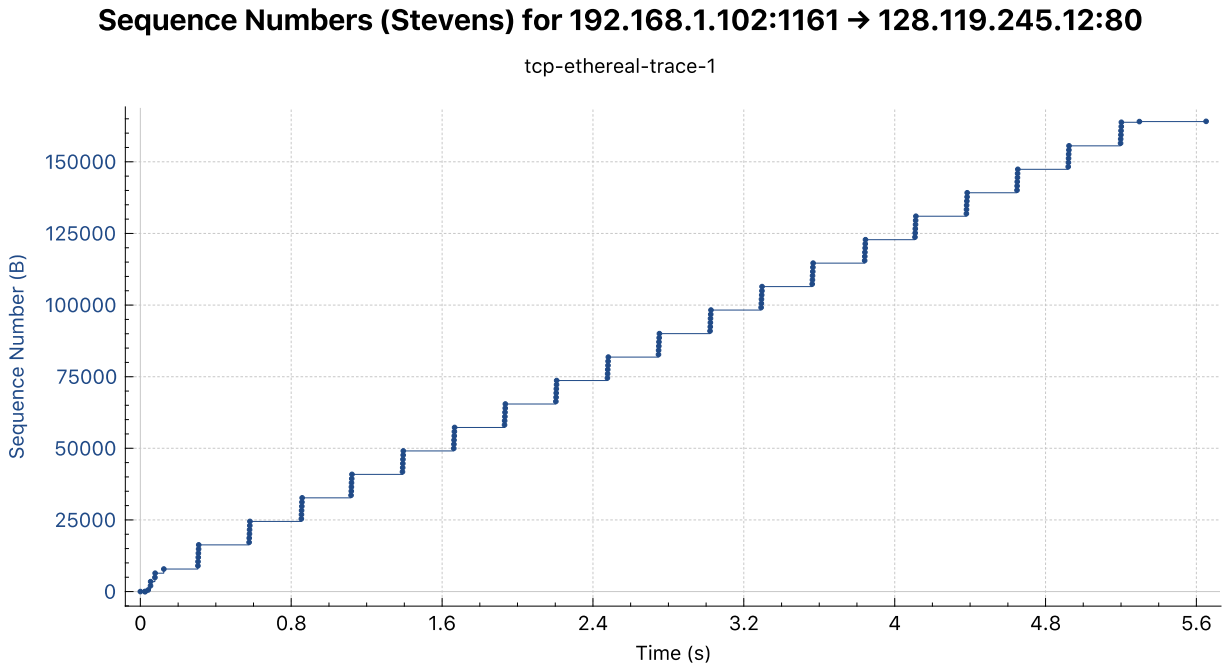
Rob Navarro
CS 372



12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.
    Answer: We can determine the throughput by determining the number of bytes transmitted and the total time it took to transmit. We then divide total transferred by time and we get our answer.
    Total bytes transferred = 164090
    Total Time = 5.455830 - .26477 = 5.4294
    Throughput = total transferred/time = 164090/5.4294 = 30222.49 bytes/sec

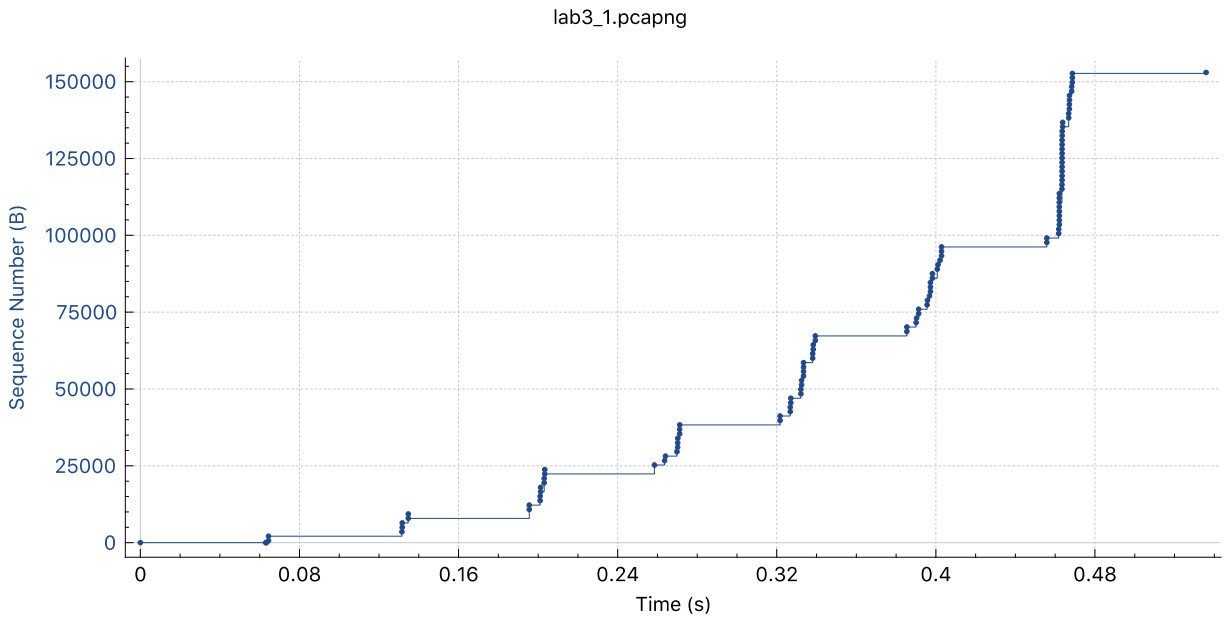
13. Use the Time-Sequence-Graph(Stevens) plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Can you identify where TCP's slowstart phase begins and ends, and where congestion avoidance takes over? Comment on ways in which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

Rob Navarro
CS 372



The graph shows that the slow start phase lasts for roughly 0.1 seconds. It then appears that the TCP session is in some sort of congestion avoidance. The weird thing is that the TCP window does not grow at all. This appears to point that the sender is not sending data aggressively or they have some sort of limit on their end.

14. Answer Question 13 for the trace that you captured when you transferred a file from your own computer to gaia.cs.umass.edu



Answer: The slow start phase appears to be working the entire time of my TCP session.

We can see this since the window size is continuously increasing during the entire transfer time. This is more of the expected behavior that we would expect to see for a transmission based on what I've read in the textbook. Since the transmission size never decreases, it does not appear that congestion control ever kicks in.