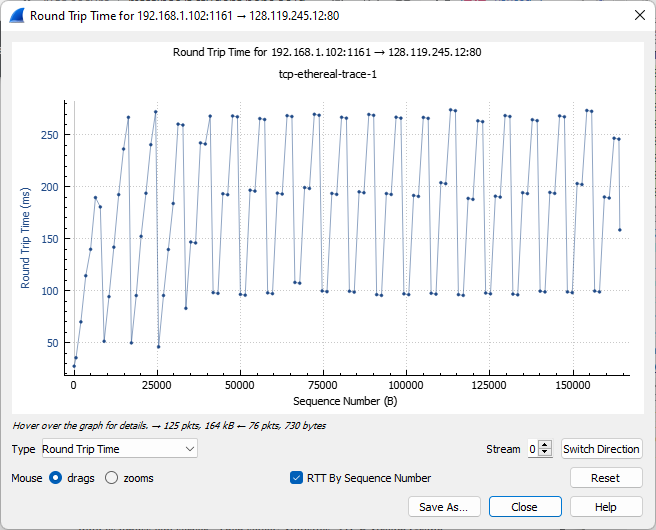
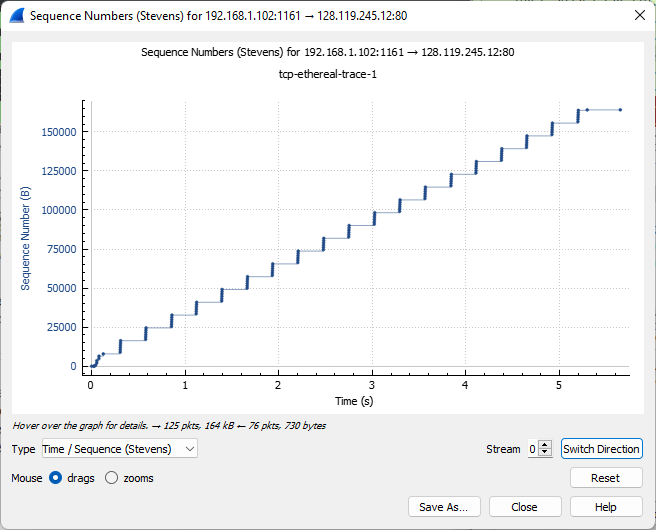
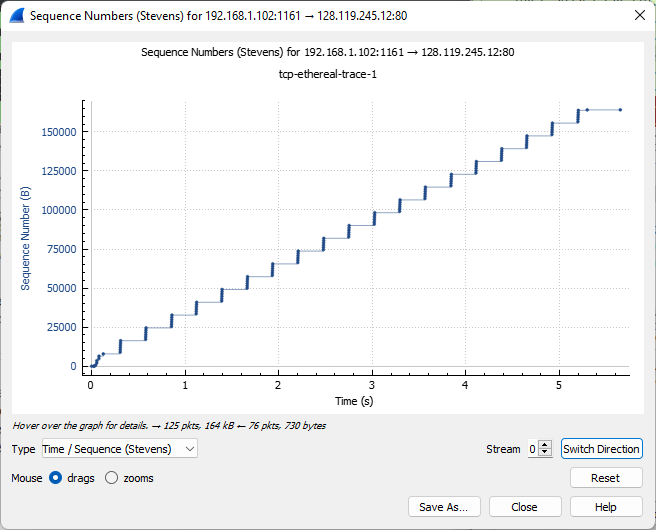
1. What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu?
   1. The IP address and TCP port number of my computer is 192.168.1.102 and the port number is 1161
2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?
   1. The IP address and port number for gaia.cs.umass.edu is 128.119.245.12 and 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?
   1. Since I am using the ethereal trace, the answer is that the IP address and TCP port number of my computer is 192.168.1.102 and the port number is 1161
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?
   1. The sequence number that is used to initiate connection is 0 in the trace. Since the SYN flag is set to 1 and the trace itself showing that it is a SYN segment in the info tab
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?
   1. The sequence number via the trace is still 0. The ACK field’s value is 1 in the SYNACK segment and that value was determined by adding one to the initial number, which in this case was 0. Since both the SYN and ACK flag values are 1, and because the info tab says it we know this is a SYNACK statement.
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.
   1. The TCP segment’s sequence number is 1 in packet 4
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 page 237 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 237 for all subsequent segments
   1. Packet 4, Sequence 1, Ack =.053937, Sent=.026477, RTT=.02746,
      1. Estimated RTT = .02746
   2. Packet 5, Sequence 566, Ack=.077294, Sent=.041737, RTT=.035557
      1. Estimated RTT = .875\*.02746+.125\*.035557=.02847
   3. Packet 7, Sequence 2026, Ack=.124085, Sent=.054026, RTT=.070059
      1. Estimated RTT = .875\*.02847+.125\*.070059=.03367
   4. Packet 8, Sequence 3486, Ack=.169118, Sent=.054690, RTT=.114428
      1. Estimated RTT = .875\*.03367+.125\*.114428=.04376
   5. Packet 10, Sequence 4946, Ack=.217299, Sent=.077405, RTT=.139894
      1. Estimated RTT = .875\*.04376+.125\*.139894=.05578
   6. Packet 11, Sequence 6406, Ack=.267802, Sent=.078157, RTT=.189645
      1. Estimated RTT = .875\*.05578+.125\*.189645=.07251
   7. 
8. What is the length of each of the first six TCP segments?
   1. First Segment-565
   2. Second Segment-1460
   3. Third Segment-1460
   4. Fourth Segment-1460
   5. Fifth Segment-1460
   6. Sixth Segment-1460
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?
   1. The minimum amount of buffer space advertised at the start is 5840, and it grows to 62780 in the end. The sender does not get throttled because there is a lacking of buffer space after looking at the trace.
10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?
    1. There are no retransmitted segments in the file, and we know that because the Time-Sequence Graph are increasing at the same rate, and if there was a retransmitted segment that would not be the case
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 247 in the text).
    1. The receiver typically receives 1460 bytes. A few examples of where the reicever ACKed every other received packet was in packets 80, 87, and 88.
12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value
    1. To calculate throughput we take the ratio between the total amount data and the total transmission time. To get the total amount data we take the sequence number of the last ACK minus the first ACK (164091-1=164090). To get the total transmission time we take the initial TCP segment minus the last ACK (5.45583-.026477= 5.429353). After calculating those two values we divide them and get 164090/5.429353=30.2275 Kbytes per second for the throughput
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.
    1. The slow start phase is from 0-.2 roughly as seen below. Congestion Avoidance is then operating for the rest of the run time. The only real way that the data given is differed from what is shown in the text is that this sequence does not drop any packets, like what we saw the text talk about a lot 
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
    1. The first question that I had for the trace was how could we actually see a dropped packet, and I now know that we can see it on Stevens Graphs
    2. The other question was how can we tell in wireshark which packets sent out belong to which packet that is ACKed and that is through looking at the info in the display area.