Name: Minh Khai Tran
zID: z5168080

Question 1:
What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection? 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?

IP address of gaia.cs.umass.edu: 128.119.245.12
Port: 1161

IP address of client computer is: 192.168.1.102
Port: 1161

Question 2:
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.

Sequence number of the TCP segment containing the HTTP POST command is:
232129013

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 web server (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.

First six sequence number:

Length	Sequence num	Sending Time	ACK received	RTT	EstimatedRTT
565 bytes	232129013	0.026477 s	0.053937 s	0.02746 s	0.02746 s
1460 bytes	232129578	0.041737 s	0.077294 s	0.035557 s	0.0285 s
1460 bytes	232131038	0.054026 s	0.124085 s	0.070059 s	0.0337 s
1460 bytes	232132498	0.054690 s	0.169118 s	0.114428 s	0.0438 s
1460 bytes	232133958	0.077405 s	0.217299 s	0.139894 s	0.0558 s
1460 bytes	232135418	0.078157 s	0.267802 s	0.189645 s	0.0725 s

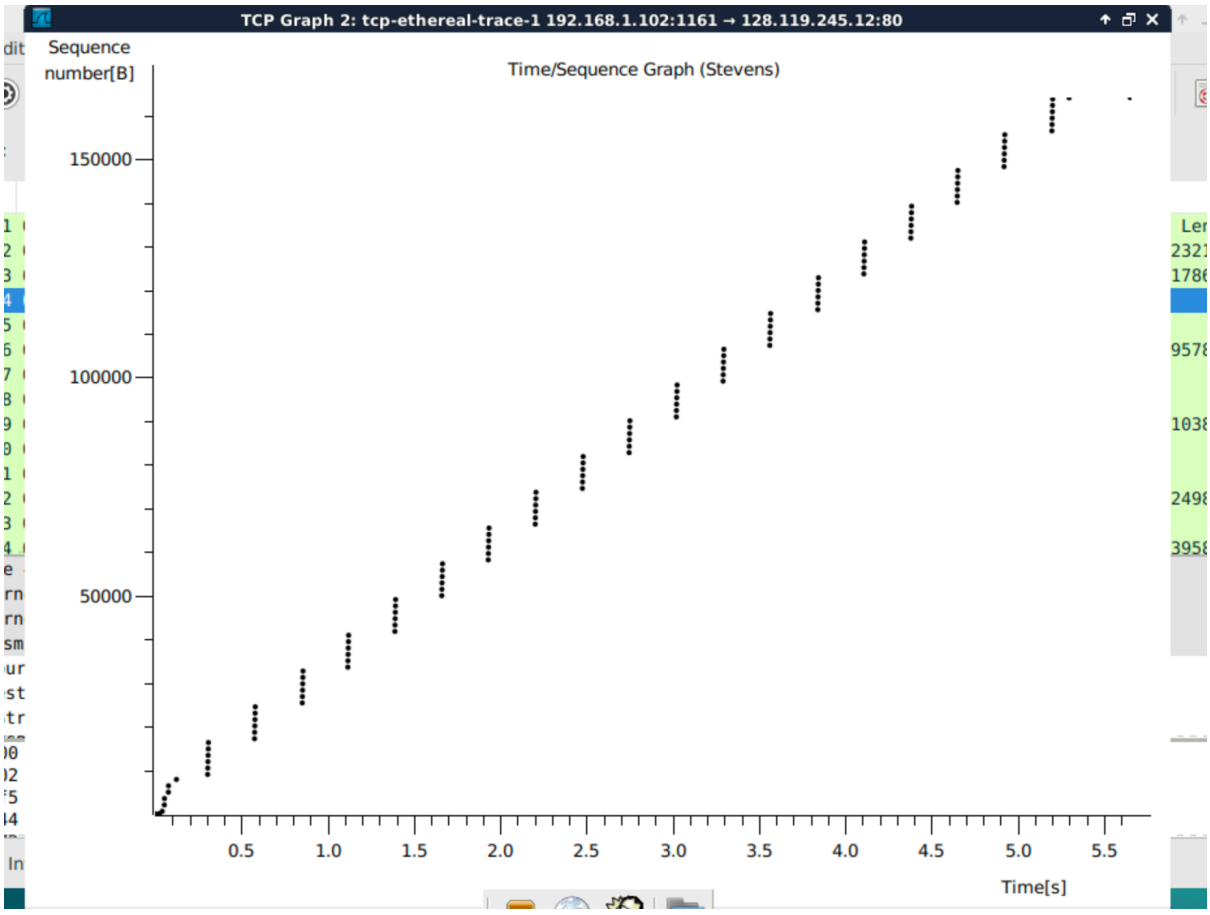
Question 4: What is the length of each of the first six TCP segments?

Seg1: 565 bytes, Seg2 = Seg3 = Seg4 = Seg5 = Seg6 =1460 bytes

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?

The minimum buffer size is 5840 bytes and the last buffer size is 62780 bytes. As we observed through the traces, the buffer size is increasing so the lack of receiver buffer space does not 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?

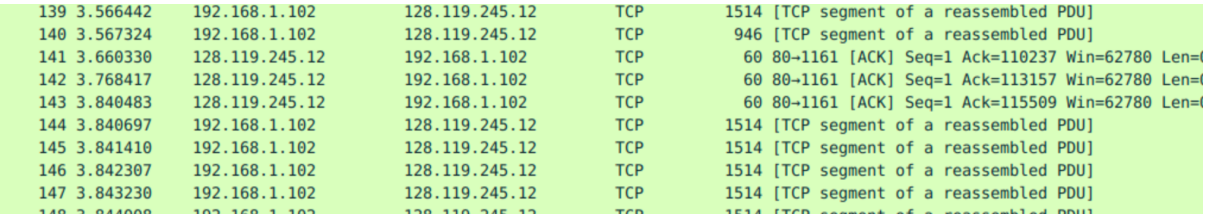


By look at the graph showing Time/Sequence (Stevens), we can tell the sequence number is always increasing, when time grows, it shows that there are NO retransmission in this trace file.

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).

Typically, the date that the receiver acknowledge in an ACK is 1460 (subtracting of two consecutive ACK number)

But at the picture below, we can see tcp segment number 141 and 142, the different of the two ACK numbers is: 113157 – 110237 = 2920 = 2*1460 = 2*(data of a typical TCP segment sent by sender). This is one of the accumulative ACK.



139	3.566442	192.168.1.102	128.119.245.12	TCP	1514 [TCP segment of a reassembled PDU]
140	3.567324	192.168.1.102	128.119.245.12	TCP	946 [TCP segment of a reassembled PDU]
141	3.660330	128.119.245.12	192.168.1.102	TCP	60 80→1161 [ACK] Seq=1 Ack=110237 Win=62780 Len=0
142	3.768417	128.119.245.12	192.168.1.102	TCP	60 80→1161 [ACK] Seq=1 Ack=113157 Win=62780 Len=0
143	3.840483	128.119.245.12	192.168.1.102	TCP	60 80→1161 [ACK] Seq=1 Ack=115509 Win=62780 Len=0
144	3.840697	192.168.1.102	128.119.245.12	TCP	1514 [TCP segment of a reassembled PDU]
145	3.841410	192.168.1.102	128.119.245.12	TCP	1514 [TCP segment of a reassembled PDU]
146	3.842307	192.168.1.102	128.119.245.12	TCP	1514 [TCP segment of a reassembled PDU]
147	3.843230	192.168.1.102	128.119.245.12	TCP	1514 [TCP segment of a reassembled PDU]

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

Last ack number from the server to client is: 232293103 at 5.455830s at TCP number 202

First sequence number from the server to client is: 232129013 at 0.026477s at TCP number 4

So throughput: $\frac{Data}{time} = \frac{232293103-232129013}{5.455830-0.026477} = \ 30222.75\ bytes/s$

$$Throughput = 241782bps$$

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?

Sequence number of the TCP SYN segment is: 2818463618

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?

Sequence number of the SYNACK segment is: 1247095790

The value of the Acknowledgement field is: 2818463619

Acknowledgement = sequence number from previous TCP segment (SYN segment) + 1

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?

Sequence number of the ACK segment is: 2818463619

The value of Acknowledgment file is: 1247095791

Acknowledgement = sequence number from the previous TCP segment (SYNACK segment) +1

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?

Both client and the server have the active close cause their ACK and SEQ is not in the format :

ACK = SEQ of the previous +1

Because the ACK number of the TCP number 305 (FINACK segment) is the same as the sequence number of the TCP number 304 (FINACK segment) so this shows that the type of this close is 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?

Ignore the 3-way handshake and the closing TCP segments:

The client starts to send data at TCP segment number 298 with sequence number is 2818463619

And at sequence number of TCP segment number 303 is the same as ACK number of TCP segment 305 which is 2818463652. It means at TCP segment number 303; the client does not send any data.

So, the total data from client to the server is: 2818463652 – 2818463619 = 33 bytes.

At TCP segment number 302, the server starts to send data to client with sequence number 1247095791.

And at TCP segment number 305, the server sends FINACK segment with sequence number 1247095831, and because this is the FINACK segment, so the server does not send any data at this time.

So, the total data from the server to client is: 1247095831 – 1247095791 = 40 bytes.

The relationship is the bytes sent.