Wireshark Lab 3: TCP

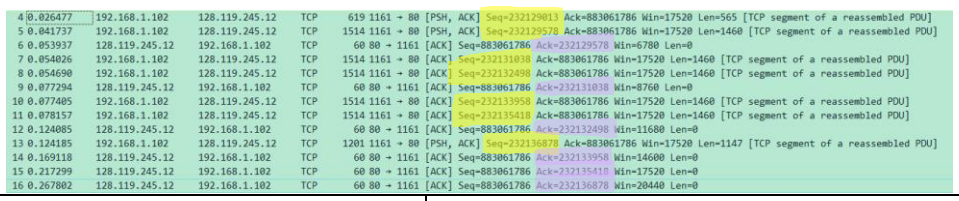
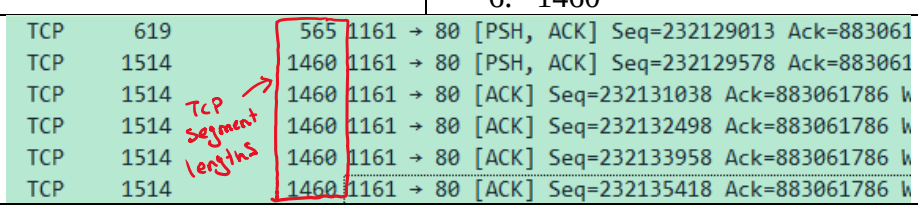
Group Details:
Shayshu NR – 1005035196
Johnathan Yan - 1004745476

Mark:

	Question	Answer
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?	Client IP: 192.168.1.102 Client source port: 1161
Annotated Screenshots (if needed)	Source Address: 192.168.1.102 Destination Address: 128.119.245.12 Transmission Control Protocol, Src Port: Source Port: 1161 Destination Port: 80 From TCP-ethereal	
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?	Server IP: 128.119.245.12 Server destination port: 80
Annotated Screenshots (if needed)	Source Address: 192.168.1.102 Destination Address: 128.119.245.12 Transmission Control Protocol, Src Port: Source Port: 1161 Destination 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?	Client IP: 10.0.0.230 Client source port: 58489

Annotated Screenshots (if needed)	Source Address: 10.0.0.230 ← My Computer Destination Address: 128.119.245.12 Transmission Control Protocol, Src Port: 58489, D Source Port: 58489 Destination Port: 80	
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?	The relative sequence number is 0 (raw 23219012). The SYN flag is set in this segment.
Annotated Screenshots (if needed)	Sequence Number: 0 (relative sequence number) Sequence Number (raw): 232129012 [Next Sequence Number: 1 (relative sequence number)] Acknowledgment Number: 0 Acknowledgment number (raw): 0 0111 .... = Header Length: 28 bytes (7) Flags: 0x002 (SYN) 000. .... = Reserved: Not set ...0 .... = Nonce: Not set .... 0... .... = Congestion Window Reduced (CWR): Not set .... .0.. .... = ECN-Echo: Not set .... ..0. .... = Urgent: Not set ....0 .... = Acknowledgment: Not set .... 0... = Push: Not set .... .0.. = Reset: Not set ....1. = Syn: Set ....0 = Fin: Not set [TCP Flags: ······· ·····S·]	
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?	The relative sequence number in the SYNACK is 0 (raw 883061785). The relative acknowledgement number is 1 (raw 23219013). This value is the sequence number that the client sent incremented by one. The SYN, and ACK flags are set thus denoting it as a SYNACK segment.

Annotated Screenshots (if needed)	Sequence Number: 0 (relative sequence number) Sequence Number (raw): 883061785 [Next Sequence Number: 1 (relative sequence number)] Acknowledgment Number: 1 (relative ack number) Acknowledgment number (raw): 232129013 0111 .... = Header Length: 28 bytes (7) Flags: 0x012 (SYN, ACK) 000. .... = Reserved: Not set ...0 .... = Nonce: Not set .... 0... .... = Congestion Window Reduced (CWR): Not set .... .0.. .... = ECN-Echo: Not set .... ..0. .... = Urgent: Not set ....1 .... = Acknowledgment: Set .... 0... = Push: Not set .... .0.. = Reset: Not set ....1. = Syn: Set ....0 = Fin: Not set [TCP Flags: ·····A··S·] Both flags set	
6	What is the sequence number of the TCP segment containing the HTTP POST command?	The relative sequence number is 1 (raw 232129013)
Annotated Screenshots (if needed)	Sequence Number: 1 (relative sequence number) Sequence Number (raw): 232129013 POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1 Host: gaia.cs.umass.edu User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.0.2) Gecko/20030208 Netscape/7.02 Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,video/x-mng, Accept-Language: en-us, en;q=0.50 Accept-Encoding: gzip, deflate, compress;q=0.9 Accept-Charset: ISO-8859-1, utf-8;q=0.66, *;q=0.66 Keep-Alive: 300 Connection: keep-alive Referer: http://gaia.cs.umass.edu/ethereal-labs/lab3-1.htm Post Request	
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	Sequence numbers: 1. 232129013 2. 232129578 3. 232131038 4. 232132498 5. 232133958 6. 232135418 Time sent: 1. 0.026477s 2. 0.041737s 3. 0.054026s 4. 0.054690s

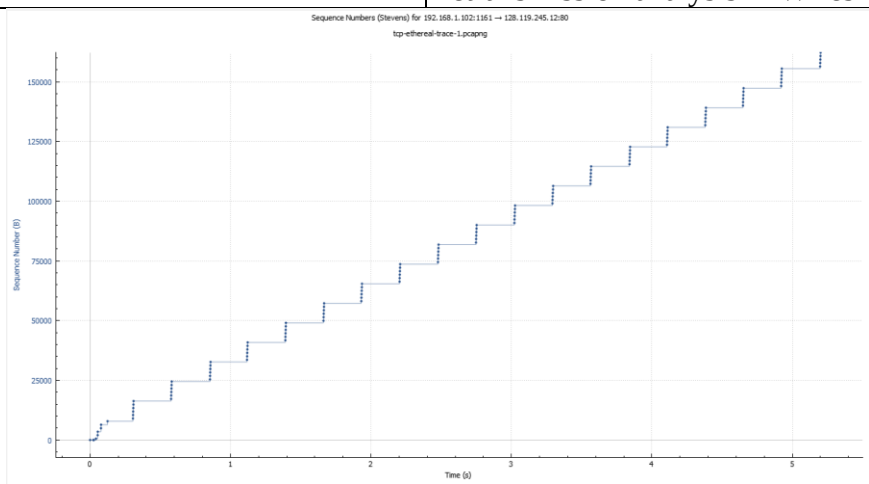
	when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments? What is the EstimatedRTT value after the receipt of each ACK?	5. 0.077405s 6. 0.078157s Time received: 1. 0.053937s 2. 0.077294s 3. 0.124085s 4. 0.169118s 5. 0.217299s 6. 0.267802s RTT: 1. 0.027460s 2. 0.035557s 3. 0.070059s 4. 0.114428s 5. 0.139894s 6. 0.190397s Les α = 0.125 Estimate RTT: 1. 0.027460s 2. 0.028472s 3. 0.033670s 4. 0.043765s 5. 0.055781s 6. 0.072608s
Annotated Screenshots (if needed)		
8	What is the length of each of the first six TCP segments?	1. 565 2. 1460 3. 1460 4. 1460 5. 1460 6. 1460
Annotated Screenshots		

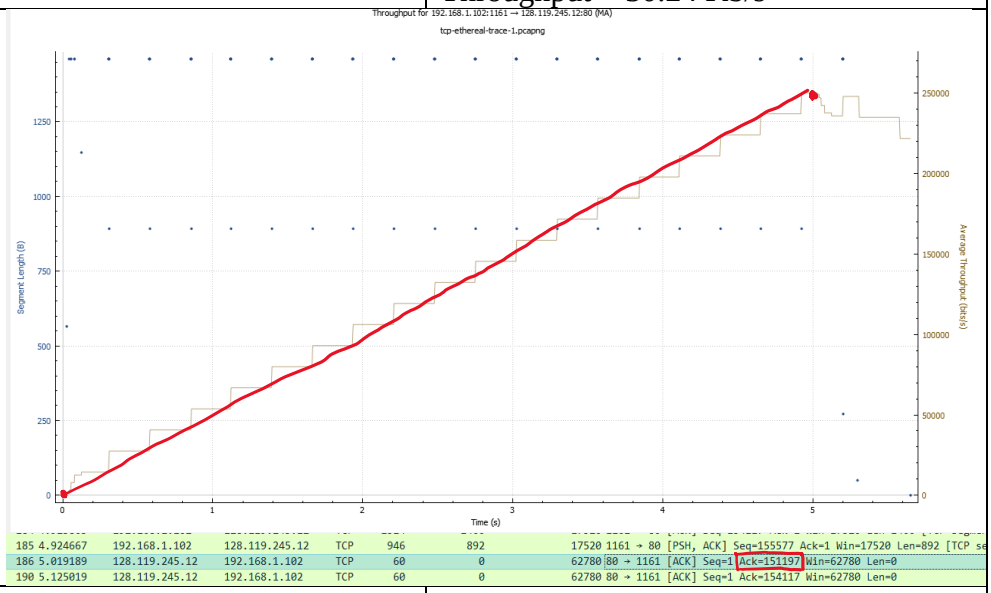
4	0.026477	192.168.1.102	128.119.245.12	TCP
5	0.041737	192.168.1.102	128.119.245.12	TCP
6	0.053937	128.119.245.12	192.168.1.102	TCP
7	0.054026	192.168.1.102	128.119.245.12	TCP
8	0.054690	192.168.1.102	128.119.245.12	TCP
9	0.077294	128.119.245.12	192.168.1.102	TCP
10	0.077405	192.168.1.102	128.119.245.12	TCP
11	0.078157	192.168.1.102	128.119.245.12	TCP
12	0.124085	128.119.245.12	192.168.1.102	TCP
13	0.124185	192.168.1.102	128.119.245.12	TCP
14	0.169118	128.119.245.12	192.168.1.102	TCP
15	0.217299	128.119.245.12	192.168.1.102	TCP
16	0.267802	128.119.245.12	192.168.1.102	TCP

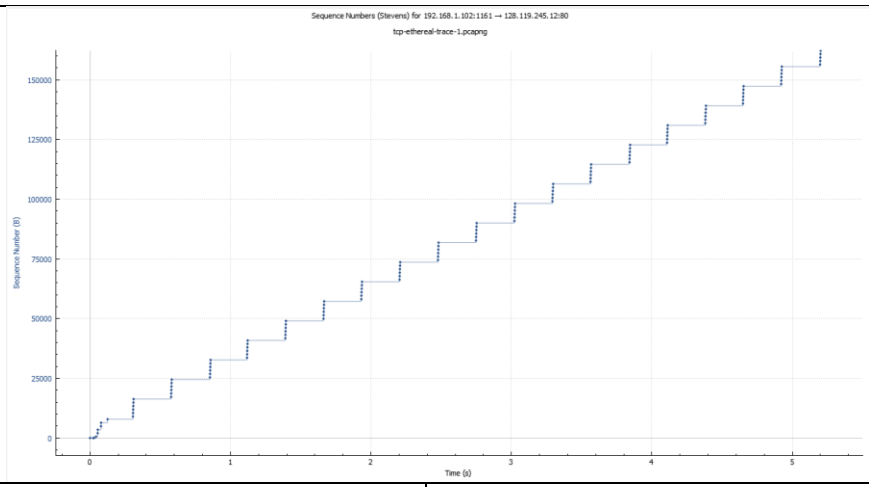
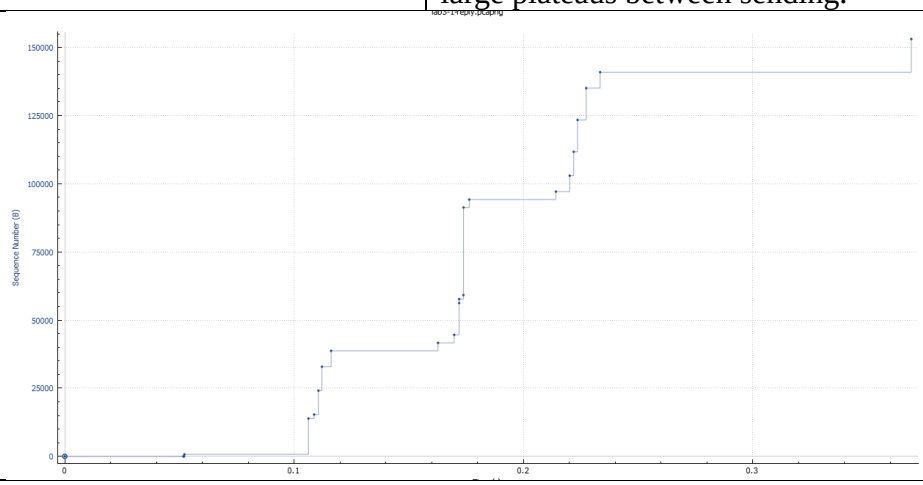
619	565	1161 → 80	[PSH, ACK]	Seq=232129013 Ack=883061786 Win=17520 Len=565 [TCP segment of a reassembled PDU]
1514	1460	1161 → 80	[PSH, ACK]	Seq=232129578 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
60	0	80 → 1161	[ACK]	Seq=883061786 Ack=232129578 Win=6780 Len=0
1514	1460	1161 → 80	[ACK]	Seq=232131038 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
1514	1460	1161 → 80	[ACK]	Seq=232132498 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
60	0	80 → 1161	[ACK]	Seq=883061786 Ack=232131038 Win=8760 Len=0
1514	1460	1161 → 80	[ACK]	Seq=232133958 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
1514	1460	1161 → 80	[ACK]	Seq=232135418 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
60	0	80 → 1161	[ACK]	Seq=883061786 Ack=232132498 Win=11680 Len=0
1201	1147	1161 → 80	[PSH, ACK]	Seq=232136878 Ack=883061786 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
60	0	80 → 1161	[ACK]	Seq=883061786 Ack=232133958 Win=14600 Len=0
60	0	80 → 1161	[ACK]	Seq=883061786 Ack=232135418 Win=17520 Len=0
60	0	80 → 1161	[ACK]	Seq=883061786 Ack=232136878 Win=20440 Len=0

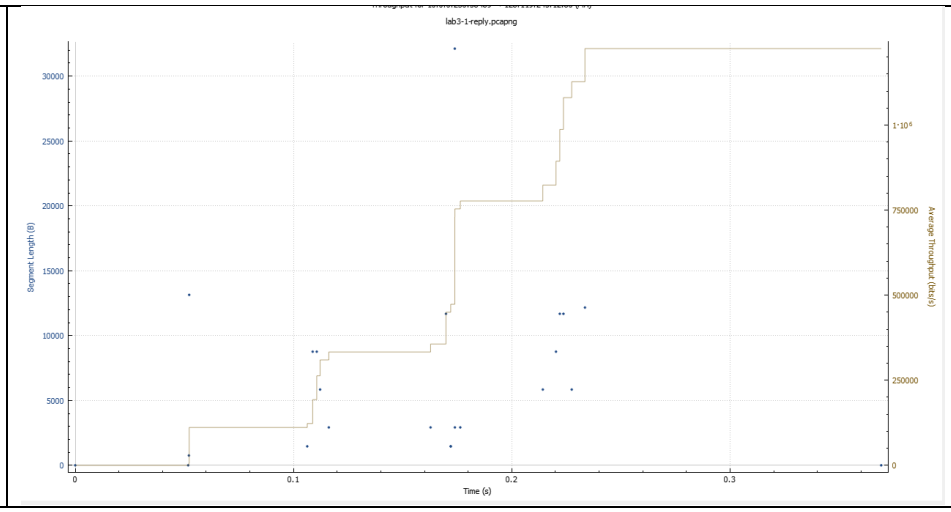
TCP	619	565	1161 → 80	[PSH, ACK] Seq=232129013 Ack=883061
TCP	1514	1460	1161 → 80	[PSH, ACK] Seq=232129578 Ack=883061
TCP	1514	1460	1161 → 80	[ACK] Seq=232131038 Ack=883061786 W
TCP	1514	1460	1161 → 80	[ACK] Seq=232132498 Ack=883061786 W
TCP	1514	1460	1161 → 80	[ACK] Seq=232133958 Ack=883061786 W
TCP	1514	1460	1161 → 80	[ACK] Seq=232135418 Ack=883061786 W

TCP segment lengths

(if needed)		
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?	Window size increases over time: 1. 5840 2. 6780 3. 8760 4. 11680 5. 14600 6. 17520 No, the window size doesn't throttle the sender. Minimum is 5840.
Annotated Screenshots (if needed)	Calculated window size / Info 5840 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1466 6780 80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0 8760 80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0 11680 80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0 14600 80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0 17520 80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0 TCP window sizes	
10	Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?	There aren't any retransmitted files. You simply look for packets with the same sequence numbers being sent at different times. Also, you can check the time vs sequence number graph, or the retransmission analysis in Wireshark.
Annotated Screenshots (if needed)		
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	The receiver usually acknowledges about 1460 bytes per ACK segment. No in the given trace, there is an acknowledgement for every segment. Meaning send 3 segments get 3 ACKs.
Annotated	17520 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reas 17520 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reas 11680 80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0	

Screenshots (if needed)		
12	What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.	Throughput is defined as the amount of data transmitted in each period of time. I chose to use a period of 5 seconds, since that is when most of the data was done being transferred. By using the sequence number and the scaling factor you can find how much data has been transmitted after 5 seconds. Throughput = 151197 bytes / 5 seconds Throughput = 30239.1 bytes / second Throughput = 30.24 Kb/s
Annotated Screenshots (if needed)		
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.	The slow start period is from 0s to about 0.1s. The Congestion avoidance starts at 0.1s and continues till the end of the transfer (around 5s). The graph is discretized and not smooth, this is because there is a wait time for the client to receive the acknowledgements from the server. Also, in this graph the client is implementing pipelining thus causing the data points to stack on top of each other.

Annotated Screenshots (if needed)		
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	Throughput = 153078 bytes/ 2.12s Throughput = 72.21 Kb/s The slow start region looks to be from 0s to 0.2s, followed by congestion avoidance from the rest. This graph shows how real pipelining works. You can clearly see the staggered sending of multiple one right after the other. Furthermore, this graph illustrates how the sender can transmit data faster than the receiver can reply, as evident by the large plateaus between sending.
Annotated Screenshots (if needed)		


lab3-1-reply.pcapng
Segment Length (B)
Average Throughput (bit/s)
Time (s)
30000
25000
20000
15000
10000
5000
0
1·10⁶
750000
500000
250000
0
0
0.1
0.2
0.3