

COMP3331 Lab4**Exercise 1: Understanding TCP using Wireshark**

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM=1

Q1. The IP address of gaia.cs.umass.edu is 128.119.245.12 using port number 80. The IP address used by the client computer is 192.168.1.102 using port number 1161.

4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565 [TCP segment of a reassembled PDU]
---	----------	---------------	----------------	-----	-----	---

00	06	25	da	af	73	00	20	e0	8a	70	1a	08	00	45	00	...	s	...	p	...	E
02	5d	1e	21	40	00	80	06	a2	e7	c0	a8	01	66	80	77	...	!	@	...	f	w
f5	0c	04	89	00	50	0d	d6	01	f5	34	a2	74	1a	50	18	...	P	...	4	t	P
44	70	1f	bd	00	00	50	4f	53	54	20	2f	65	74	68	65	Dp	...	P	O	ST	/ethe

Q2. The sequence number of the HTTP POST command is 232129013.

4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565 [TCP segment of a reassembled PDU]
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=232129578 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232129578 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232131038 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232132498 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
9	0.077294	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232131038 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232133958 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232135418 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
12	0.124085	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232132498 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=232136878 Ack=883061786 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
14	0.169118	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232133958 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232135418 Win=17520 Len=0
16	0.267802	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232136878 Win=20440 Len=0

Q3a. The sequence numbers of the first six segments (from POST) being sent from the client are 23212901 (#4), 232129578 (#5), 232131038 (#7), 232132498 (#8), 232133958 (#10), and 232135418 (#11).

Q3b. They were sent at 0.026477, 0.041737, 0.054026, 0.054690, 0.077405, and 0.078157 seconds. The ACKs were received at 0.053937 (#6), 0.077294 (#9), 0.124085 (#12), 0.169118 (#14), 0.217299 (#15), and 0.267802 (#16) seconds. The RTTs are 0.027460, 0.035557, 0.070059, 0.114428, 0.139894, and 0.189645 seconds.

Q3c.

After receiving ACK for the first segment (#4): $(1-0.125)*0.027460 + (0.125)*0.027460 = 0.027460$ seconds.

After receiving ACK for the second segment (#5): $(1-0.125)*0.02746 + (0.125)*0.035557 = 0.028472$ seconds.

After receiving ACK for the third segment (#7): $(1-0.125)*0.028472 + (0.125)*0.070059 = 0.033670$ seconds.

After receiving ACK for the fourth segment (#8): $(1-0.125)*0.033670 + (0.125)*0.114428 = 0.043764$ seconds.

After receiving ACK for the fifth segment (#10): $(1-0.125)*0.043764 + (0.125)*0.139894 = 0.055780$ seconds.

After receiving ACK for the sixth segment (#11): $(1-0.125)*0.055780 + (0.125)*0.189645 = 0.072513$ seconds.

Q4. The length is 565, 1460, 1460, 1460, 1460, and 1460 bytes.

1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=883061785 Ack=232129013 Win=5840 Len=0 MSS=1460 SACK_PERM=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=232129013 Ack=883061786 Win=17520 Len=0

Q5. The minimum amount of available buffer space advertised at the receiver is 5840 bytes. I believe it doesn't throttle the sender since it is then increased to a maximum of 62780 bytes and nothing ever exceeds that.

tcp.analysis.retransmission or tcp.analysis.fast_retransmission						
No.	Time	Source	Destination	Protocol	Length	Info

Q6. There are no retransmitted segments using tcp.analysis.retransmission or tcp.analysis.retransmission or tcp.analysis.fast_retransmission to filter only retransmitted segments, however, nothing shows up.

Q7.

53	1.117333	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232162601 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
54	1.118133	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232164061 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
55	1.119029	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232165521 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
56	1.119858	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232166981 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
57	1.120902	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=232168441 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
58	1.121891	192.168.1.102	128.119.245.12	TCP	946	1161 → 80 [PSH, ACK] Seq=232169901 Ack=883061786 Win=17520 Len=892 [TCP segment of a reassembled PDU]
59	1.200421	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232164061 Win=62780 Len=0
60	1.265026	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232166981 Win=62780 Len=0
61	1.362074	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232169901 Win=62780 Len=0
62	1.389886	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232170793 Win=62780 Len=0

The data is normally 1460 bytes. The first case I noticed is when an ACK of 232165521 in response to #53 with SEQ=232162601 with len=1460 is skipped between #59 (ACK=232164061) and #60 (ACK=232166981), and then between #60 and #61, and more afterward. I noticed that the ACKs of the responses is not incremented according to the length of the packets sent due to delayed ACK.

Q8.

1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=883061785 Ack=232129013 Win=5840 Len=0 MSS=1460 SACK_PERM=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=232129013 Ack=883061786 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565 [TCP segment of a reassembled PDU]
200	5.389471	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232291321 Win=62780 Len=0
201	5.447887	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232293053 Win=62780 Len=0
202	5.455830	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232293103 Win=62780 Len=0
203	5.461175	128.119.245.12	192.168.1.102	HTTP	784	HTTP/1.1 200 OK (text/html)
206	5.651141	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=232293103 Ack=883062516 Win=16790 Len=0

The total amount of data = 232293103 (#202 ACK) – 232129013 (#4 SEQ) = 164090 bytes. (This is the final ACK from the last packet where len = 0 from the receiver and the SEQ of the first packet where len > 0)

The total transmission time = 5.455830 (#202 Timestamp) – 0.026477 (#4 Timestamp) = 5.429353 seconds.

The throughput = The total amount of data/The total transmission time = 164090/5.429353 = 30222.75398 bytes/second.

Exercise 2: TCP Connection Management

Q1. The sequence number is 2818463618.

Q2. The sequence number is 1247095790 and the ACK is 2818463619, this is from the sequence number from the client plus 1 byte.

Q3. The sequence number is 2818463619 and the ACK is 1247095791, this doesn't contain any data since no additional bytes were added.

Q4. Both sides have done the active close since the client and the server started sending the TCP segment with the FIN bit. It is Simultaneous Closure with ACK from both sides after the FIN bit segments.

Q5. 33 (35-2) bytes from client to server and 40 (42-2) bytes from server to client. They are the difference between the initial sequence number and the final ACK received from the other side subtracting by bits from SYN and FIN.