ASSIGNMENT 4
Wireshark for Transport Layer Protocols

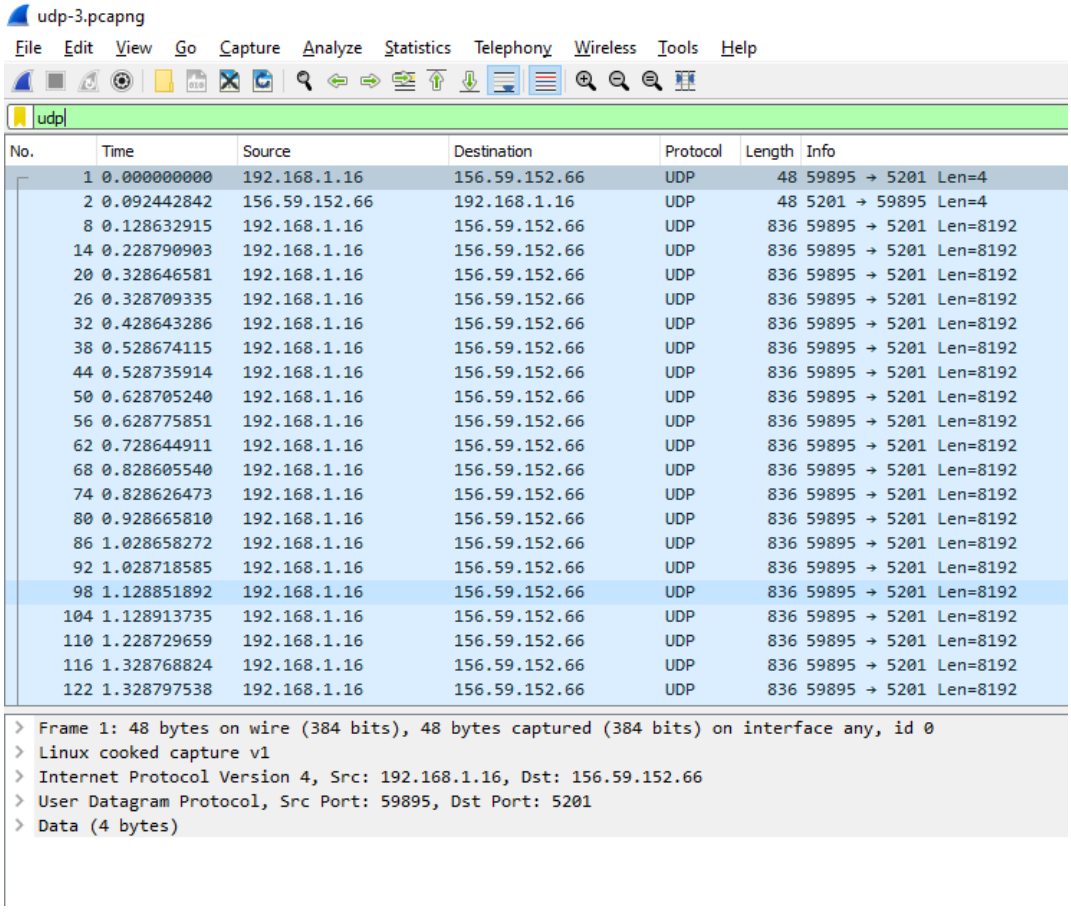
Submitted By,
SHRUSTI
CS22MTECH11017

TASK1:

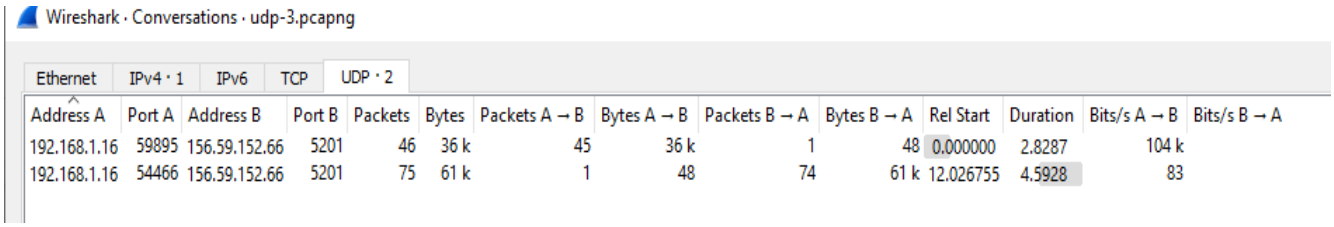
Use the attached task1-udp3.pcap file to answer the below questions. Please use udp as the display filter in the wireshark once you open it for answering the following questions.

1. How many communications/conversations are present ? List the total number of packets exchanged for each communication.

Solution:
Use “udp” filter to filter out the UDP packets that are exchanged between the endsystems.



Observing Conversations statistics we can see **2** UDP and **1 IPV4** communications/conversations are present.


Wireshark · Conversations · udp-3.pcapng

Ethernet | IPv4 · 1 | IPv6 | TCP | UDP · 2

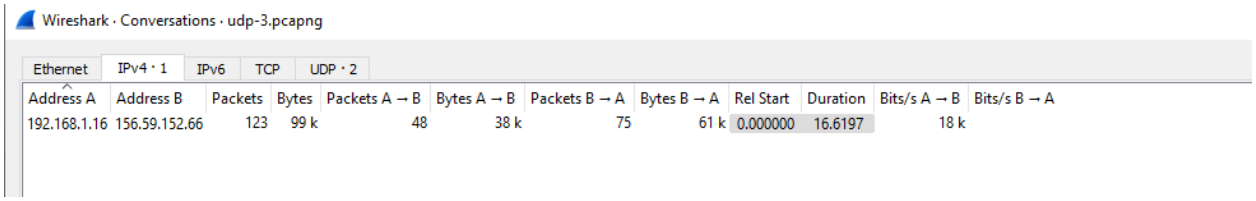
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
192.168.1.16	59895	156.59.152.66	5201	46	36 k	45	36 k	1	48	0.000000	2.8287	104 k	
192.168.1.16	54466	156.59.152.66	5201	75	61 k	1	48	74	61 k	12.026755	4.5928	83	

<u>UDP Conversation 1:</u>

Number of packets exchanged between 192.168.1.16/59895 and 156.59.152.66/5201 : **46 packets**

<u>UDP Conversation 2:</u>

Number of packets exchanged between 192.168.1.16/54466 and 156.59.152.66/5201: **75 packets**


Wireshark · Conversations · udp-3.pcapng

Ethernet | IPv4 · 1 | IPv6 | TCP | UDP · 2

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
192.168.1.16	156.59.152.66	123	99 k	48	38 k	75	61 k	0.000000	16.6197	18 k	

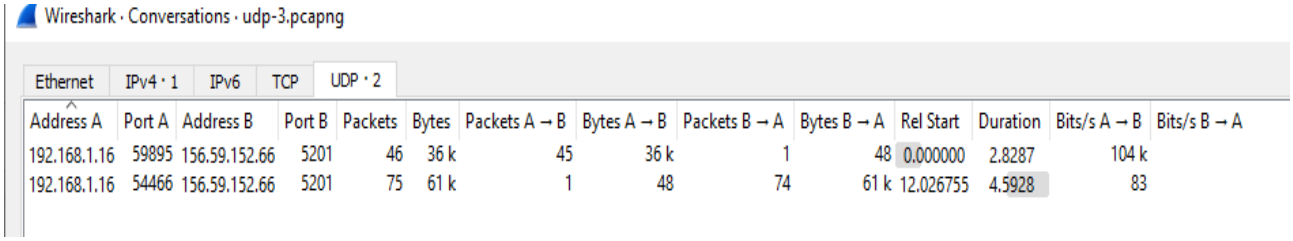
<u>IPV4 Conversation:</u>

Number of packets exchanged between 192.168.1.16 and 156.59.152.66: **123 packets**

2. Who is sending data to whom ? What is the average size of the packet sent ? Answer these questions for each of the conversations present.

Solution:

We use Conversation statistics to find the data transferred.


Wireshark · Conversations · udp-3.pcapng

Ethernet | IPv4 · 1 | IPv6 | TCP | UDP · 2

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
192.168.1.16	59895	156.59.152.66	5201	46	36 k	45	36 k	1	48	0.000000	2.8287	104 k	
192.168.1.16	54466	156.59.152.66	5201	75	61 k	1	48	74	61 k	12.026755	4.5928	83	

<u>Conversation 1 between 192.168.1.16/59895 and 156.59.152.66/5201:</u>

We see that **36k** bytes are transferred from A to B whereas **48 bytes** are transferred from B to A .

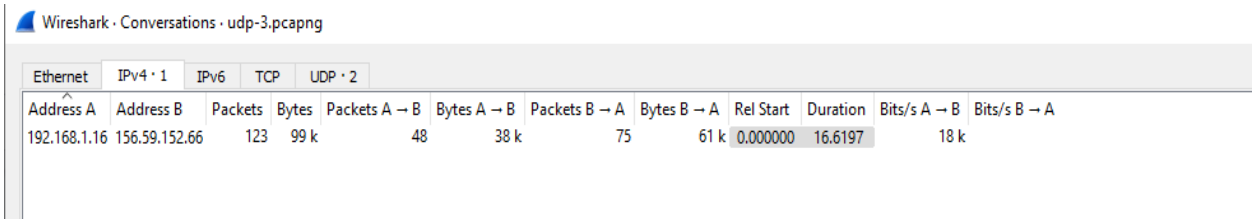
The size of the packet from A to B : (36*1024)/45 = **819.2 bytes**
The size of the packet from B to A : 48/1 = **48 bytes**

Conversation 2 between 192.168.1.16/54466 and 156.59.152.66/5201 :

We see that **48 bytes** are transferred from A to B whereas **61k bytes** are transferred from B to A .

The size of the packet from A to B : 48/1 = **48 bytes**
The size of the packet from B to A : (61*1024)/74 = **844.108 bytes**

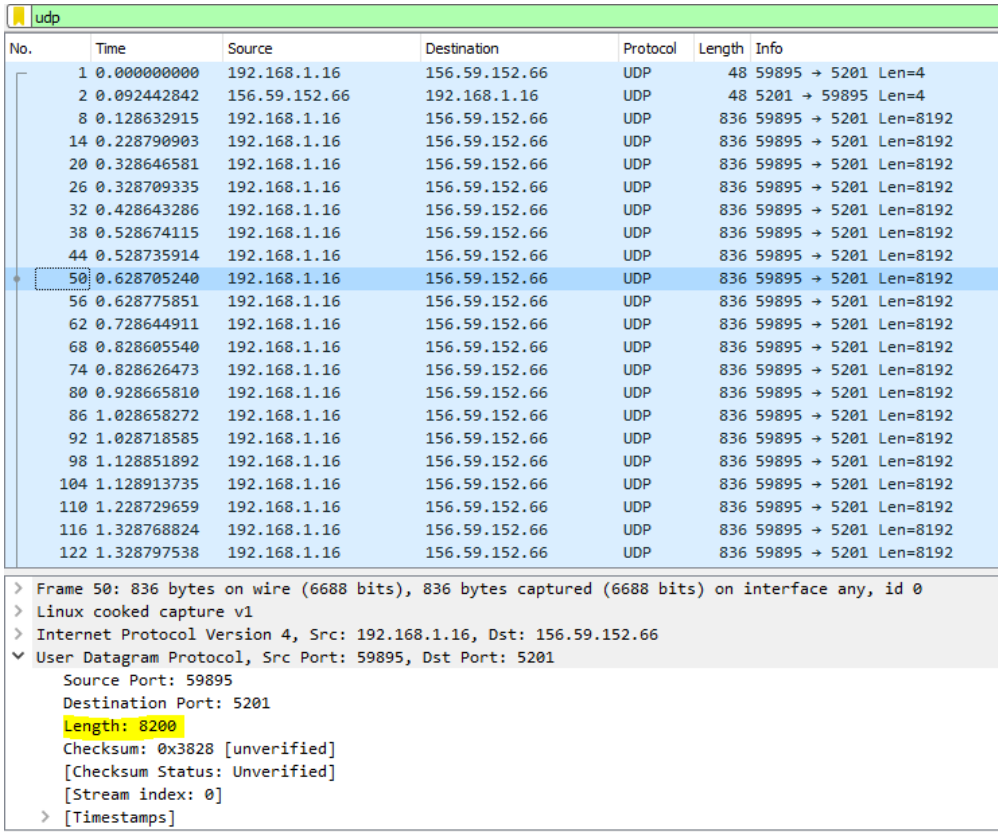
IPv4 Conversation between 192.168.1.16 and 156.59.152.66:



We see that **38k bytes** are transferred from A to B whereas **61k bytes** are transferred from B to A.

The size of the packet from A to B : (38*1024)/48 = **810.66 bytes**
The size of the packet from B to A : (61*1024)/75 = **832.85 bytes**

3. Pick any one UDP conversation (not just one packet). Calculate the throughput (bytes transferred per unit time) for UDP conversation using UDP's length field. Explain how you calculated this value using Wireshark capture in this experiment along with relevant screenshots. Verify your calculation with the one done by Wireshark using "Capture File properties" If you observe the major difference in your calculation and with the other two listed here, comment why and how ?

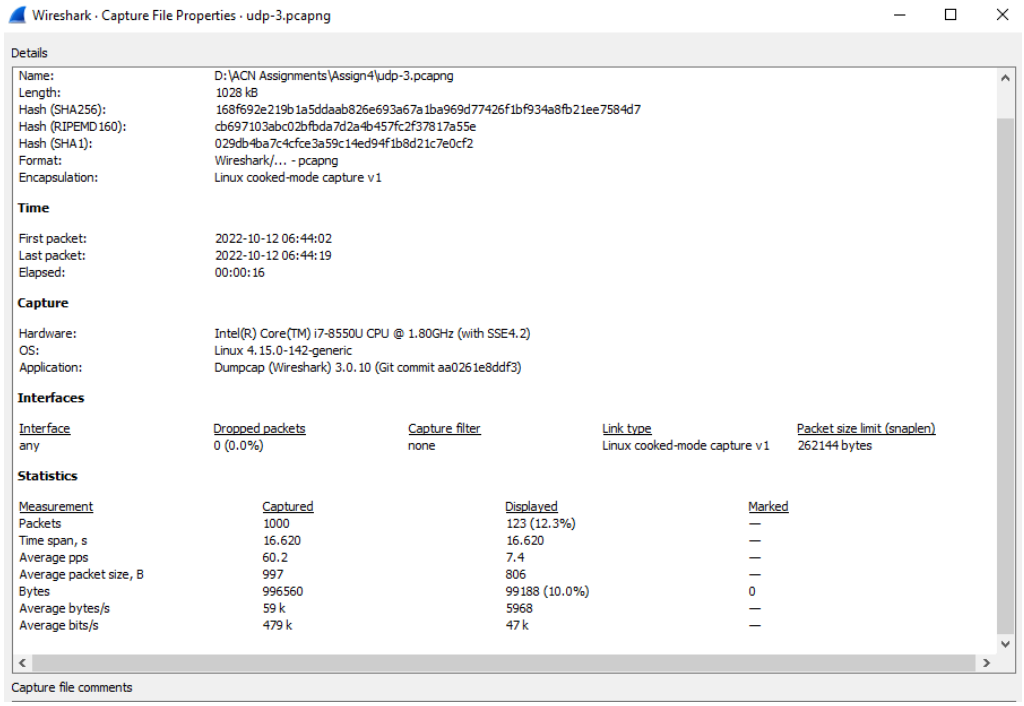


Select a UDP packet capture we can see the length of each packet is 8192 bytes(excluding header) and observe UDP Conversation 1 we had 48 packets transferred.

Total data transferred is **8192* 45 + 1*4 = 368644 bytes.**

The time required for this transmission is **2.8287 seconds** (defined in the Duration column of the UDP Conversation 1).

Therefore, the throughput is 368644 /2.8287 = **130322.763107 bytes/sec**



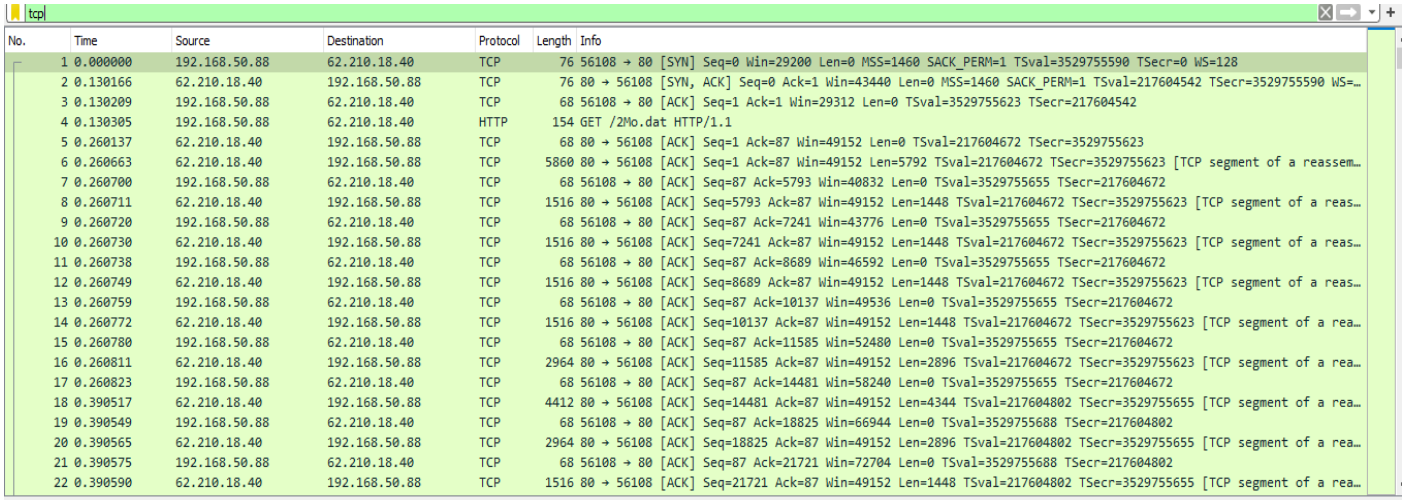
There is a major difference between the throughput calculated and throughput observed from Capture file properties because Wireshark uses length 836 i.e; the actual data of the packet whereas we use length 8192 i.e; total UDP packet length.

TASK 2

Use the attached task2-tcp3.pcap file to answer the below questions. Please use tcp as the display filter in the wireshark once you open it for answering the following questions.

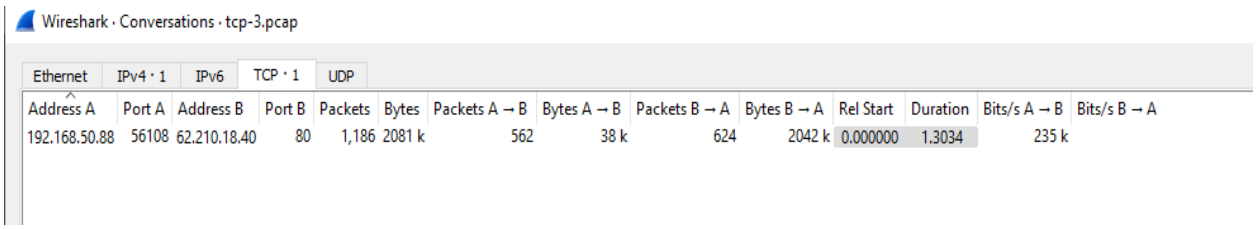
1. How many TCP connections are established ?
Solution:

Use “tcp” filter to filter out the TCP packets that are exchanged between the end systems.



tcp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.50.88	62.210.18.40	TCP	76	56108 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3529755590 TSecr=0 WS=128
2	0.130166	62.210.18.40	192.168.50.88	TCP	76	80 → 56108 [SYN, ACK] Seq=0 Ack=1 Win=43440 Len=0 MSS=1460 SACK_PERM=1 TSval=217604542 TSecr=3529755590 WS=…
3	0.130209	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3529755623 TSecr=217604542
4	0.130305	192.168.50.88	62.210.18.40	HTTP	154	GET /2Mo.dat HTTP/1.1
5	0.260137	62.210.18.40	192.168.50.88	TCP	68	80 → 56108 [ACK] Seq=1 Ack=87 Win=49152 Len=0 TSval=217604672 TSecr=3529755623
6	0.260663	62.210.18.40	192.168.50.88	TCP	5860	80 → 56108 [ACK] Seq=1 Ack=87 Win=49152 Len=5792 TSval=217604672 TSecr=3529755623 [TCP segment of a reassem…
7	0.260700	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=5793 Win=40832 Len=0 TSval=3529755655 TSecr=217604672
8	0.260711	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=5793 Ack=87 Win=49152 Len=1448 TSval=217604672 TSecr=3529755623 [TCP segment of a reas…
9	0.260720	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=7241 Win=43776 Len=0 TSval=3529755655 TSecr=217604672
10	0.260730	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=7241 Ack=87 Win=49152 Len=1448 TSval=217604672 TSecr=3529755623 [TCP segment of a reas…
11	0.260738	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=8689 Win=46592 Len=0 TSval=3529755655 TSecr=217604672
12	0.260749	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=8689 Ack=87 Win=49152 Len=1448 TSval=217604672 TSecr=3529755623 [TCP segment of a reas…
13	0.260759	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=10137 Win=49536 Len=0 TSval=3529755655 TSecr=217604672
14	0.260772	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=10137 Ack=87 Win=49152 Len=1448 TSval=217604672 TSecr=3529755623 [TCP segment of a rea…
15	0.260780	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=11585 Win=52480 Len=0 TSval=3529755655 TSecr=217604672
16	0.260811	62.210.18.40	192.168.50.88	TCP	2964	80 → 56108 [ACK] Seq=11585 Ack=87 Win=49152 Len=2896 TSval=217604672 TSecr=3529755623 [TCP segment of a rea…
17	0.260823	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=14481 Win=58240 Len=0 TSval=3529755655 TSecr=217604672
18	0.390517	62.210.18.40	192.168.50.88	TCP	4412	80 → 56108 [ACK] Seq=14481 Ack=87 Win=49152 Len=4344 TSval=217604802 TSecr=3529755655 [TCP segment of a rea…
19	0.390549	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=18825 Win=66944 Len=0 TSval=3529755688 TSecr=217604802
20	0.390565	62.210.18.40	192.168.50.88	TCP	2964	80 → 56108 [ACK] Seq=18825 Ack=87 Win=49152 Len=2896 TSval=217604802 TSecr=3529755655 [TCP segment of a rea…
21	0.390575	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=21721 Win=72704 Len=0 TSval=3529755688 TSecr=217604802
22	0.390590	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=21721 Ack=87 Win=49152 Len=1448 TSval=217604802 TSecr=3529755655 [TCP segment of a rea…



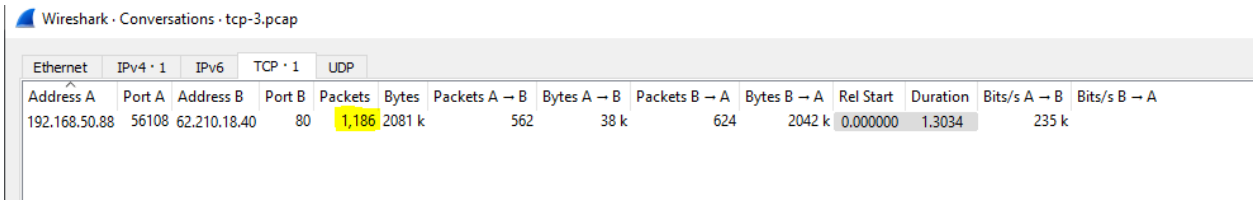
Wireshark · Conversations · tcp-3.pcap

Ethernet | IPv4 · 1 | IPv6 | TCP · 1 | UDP

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
192.168.50.88	56108	62.210.18.40	80	1,186	2081 k	562	38 k	624	2042 k	0.000000	1.3034	235 k	

From Statistics → Conversations, we can see that only **1 TCP** connection is established.

2. How many TCP packets are exchanged in this communication client and remote server?



Wireshark · Conversations · tcp-3.pcap

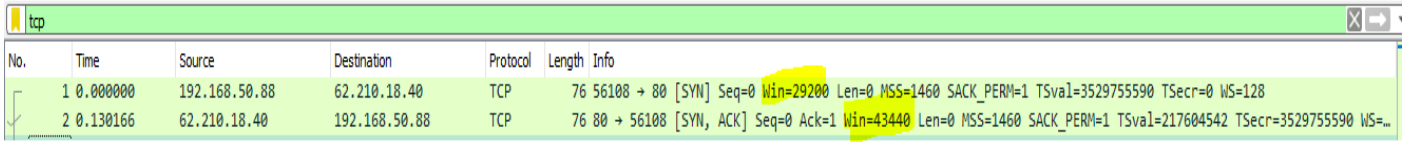
Ethernet | IPv4 · 1 | IPv6 | TCP · 1 | UDP

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
192.168.50.88	56108	62.210.18.40	80	1,186	2081 k	562	38 k	624	2042 k	0.000000	1.3034	235 k	

From Statistics → Conversations, we can see **1186 packets** are exchanged in this communication.

3. What is the amount of available buffer space advertised in the beginning of the session at the client/receiver? How much does it differ from the one advertised/available during the last 10-5ms duration of the session (in the entire trace captured).

Solution:



tcp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.50.88	62.210.18.40	TCP	76	56108 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3529755590 TSecr=0 WS=128
2	0.130166	62.210.18.40	192.168.50.88	TCP	76	80 → 56108 [SYN, ACK] Seq=0 Ack=1 Win=43440 Len=0 MSS=1460 SACK_PERM=1 TSval=217604542 TSecr=3529755590 WS=…

<u>At the beginning of the session :</u>

29200 bytes is the available buffer space advertised from the client which can be seen by the SYN packet.

43440 bytes is the available buffer space advertised from the server which can be seen by the SYN+ ACK packet.

Towards end of the session:

1155	1.172667	192.168.50.88	62.210.18.40	TCP	68 56108 → 80 [ACK] Seq=87 Ack=1950457 Win=2138496 Len=0 TSval=3529755883 TSecr=217605584
1156	1.172710	62.210.18.40	192.168.50.88	TCP	2964 80 → 56108 [ACK] Seq=1950457 Ack=87 Win=49152 Len=2896 TSval=217605584 TSecr=3529755851 [TCP segment of a r…

2138496 bytes is available buffer space advertised from client and **49152 bytes** is the available buffer space advertised by the server.

The difference between the buffer size is:

Client buffer space = **2138496 - 29200 = 2109296 bytes**

Server buffer space = **49152 - 43440 = 5712 bytes**

Observing the difference we can say that buffer size is increased.

4. Who is sending the data to whom ? How many data-containing TCP segments were needed to complete the process of sending the total data ?

Solution:

ip.src == 62.210.18.40 && tcp

No.	Time	Source	Destination	Protocol	Length	Info
1136	1.172195	62.210.18.40	192.168.50.88	TCP	4412	80 → 56108 [ACK] Seq=1908465 Ack=87 Win=49152 Len=4344 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1138	1.172223	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1912809 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1140	1.172290	62.210.18.40	192.168.50.88	TCP	7308	80 → 56108 [ACK] Seq=1914257 Ack=87 Win=49152 Len=7240 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1142	1.172343	62.210.18.40	192.168.50.88	TCP	2964	80 → 56108 [ACK] Seq=1921497 Ack=87 Win=49152 Len=2896 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1144	1.172411	62.210.18.40	192.168.50.88	TCP	7308	80 → 56108 [ACK] Seq=1924393 Ack=87 Win=49152 Len=7240 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1146	1.172460	62.210.18.40	192.168.50.88	TCP	2964	80 → 56108 [ACK] Seq=1931633 Ack=87 Win=49152 Len=2896 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1148	1.172488	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1934529 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1150	1.172509	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1935977 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1152	1.172576	62.210.18.40	192.168.50.88	TCP	7308	80 → 56108 [ACK] Seq=1937425 Ack=87 Win=49152 Len=7240 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1154	1.172660	62.210.18.40	192.168.50.88	TCP	5860	80 → 56108 [ACK] Seq=1944665 Ack=87 Win=49152 Len=5792 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1156	1.172710	62.210.18.40	192.168.50.88	TCP	2964	80 → 56108 [ACK] Seq=1950457 Ack=87 Win=49152 Len=2896 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1158	1.172737	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1953353 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1160	1.172765	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1954801 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1162	1.172788	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1956249 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1164	1.172827	62.210.18.40	192.168.50.88	TCP	2964	80 → 56108 [ACK] Seq=1957697 Ack=87 Win=49152 Len=2896 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1166	1.172850	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1960593 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1168	1.172920	62.210.18.40	192.168.50.88	TCP	7308	80 → 56108 [ACK] Seq=1962041 Ack=87 Win=49152 Len=7240 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1170	1.172988	62.210.18.40	192.168.50.88	TCP	4412	80 → 56108 [ACK] Seq=1969281 Ack=87 Win=49152 Len=4344 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1172	1.173205	62.210.18.40	192.168.50.88	TCP	4412	80 → 56108 [ACK] Seq=1973625 Ack=87 Win=49152 Len=4344 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1174	1.173224	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=1977969 Ack=87 Win=49152 Len=1448 TSval=217605584 TSecr=3529755851 [TCP segment of a r…
1176	1.173295	62.210.18.40	192.168.50.88	TCP	5860	80 → 56108 [ACK] Seq=1979417 Ack=87 Win=49152 Len=5792 TSval=217605584 TSecr=3529755851 [TCP segment of a r…

Wireshark · Conversations · tcp-3.pcap

Ethernet | IPv4 · 1 | IPv6 | TCP · 1 | UDP

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
62.210.18.40	80	192.168.50.88	56108	624	2042 k	624	2042 k	0	0	0.130166	1.1732	13 M	

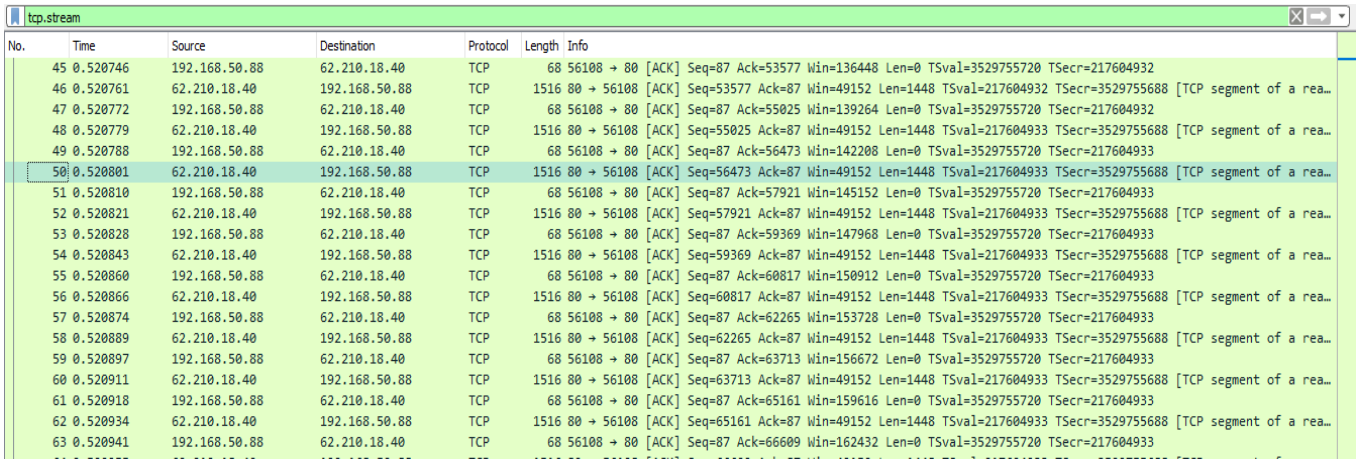
As 62.210.18.40 is sending the data ,apply the filter as **“ip.src == 62.210.18.40 && tcp “**and observe Statistics → Conversations to see the number of TCP packets by limiting the display filter.

We can see **624 TCP packets** are present and then we subtract the 4 packets which are non-data packets i.e; 2 packets are of TCP establishment and 2 packets are of TCP termination are ignored.

Therefore, 624 - 4**= 620 TCP segments** are needed to complete the process of sending the total data .

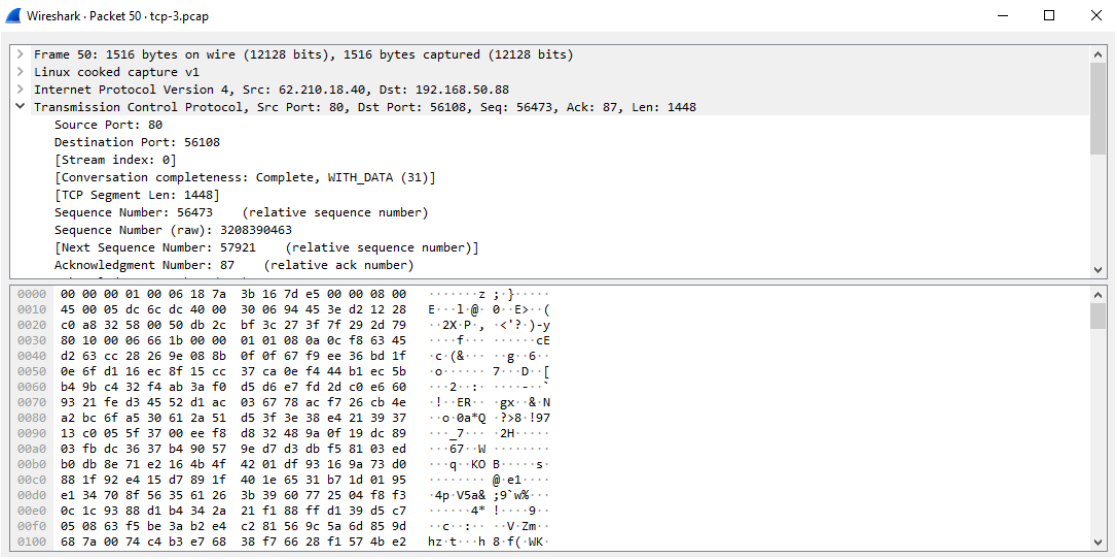
5. Pick any 5 TCP segments from server to client which are not part of initial TCP connection establishment and final connection termination.

5.1. Make a table listing for each of these segments, the length of each of these TCP segments, the sequence number, time when the segment was sent, time when the respective ACK for each segment was received, length of the respective ACK segment. Place the screenshot of Wireshark of at least one such segment with respective ACK as a proof of observation and calculation.
What is the maximum length out of all ?

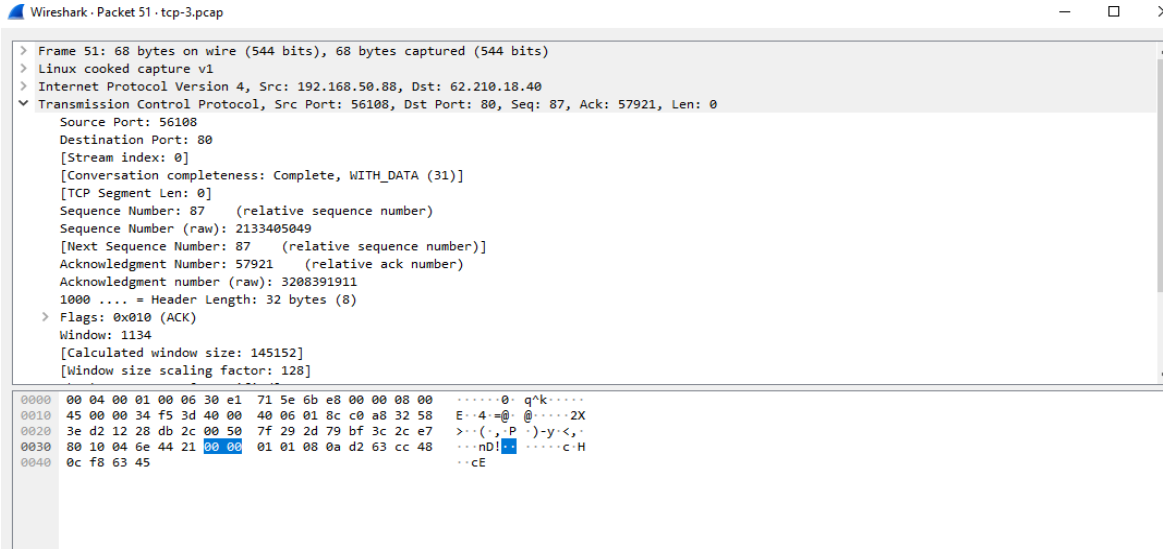

tcp.stream

No.	Time	Source	Destination	Protocol	Length	Info
45	0.520746	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=53577 Win=136448 Len=0 TSval=3529755720 TSecr=217604932
46	0.520761	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=53577 Ack=87 Win=49152 Len=1448 TSval=217604932 TSecr=3529755688 [TCP segment of a rea…
47	0.520772	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=55025 Win=139264 Len=0 TSval=3529755720 TSecr=217604932
48	0.520779	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=55025 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
49	0.520788	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=56473 Win=142208 Len=0 TSval=3529755720 TSecr=217604933
50	0.520801	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=56473 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
51	0.520810	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=57921 Win=145152 Len=0 TSval=3529755720 TSecr=217604933
52	0.520821	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=57921 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
53	0.520828	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=59369 Win=147968 Len=0 TSval=3529755720 TSecr=217604933
54	0.520843	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=59369 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
55	0.520860	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=60817 Win=150912 Len=0 TSval=3529755720 TSecr=217604933
56	0.520866	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=60817 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
57	0.520874	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=62265 Win=153728 Len=0 TSval=3529755720 TSecr=217604933
58	0.520889	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=62265 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
59	0.520897	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=63713 Win=156672 Len=0 TSval=3529755720 TSecr=217604933
60	0.520911	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=63713 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
61	0.520918	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=65161 Win=159616 Len=0 TSval=3529755720 TSecr=217604933
62	0.520934	62.210.18.40	192.168.50.88	TCP	1516	80 → 56108 [ACK] Seq=65161 Ack=87 Win=49152 Len=1448 TSval=217604933 TSecr=3529755688 [TCP segment of a rea…
63	0.520941	192.168.50.88	62.210.18.40	TCP	68	56108 → 80 [ACK] Seq=87 Ack=66609 Win=162432 Len=0 TSval=3529755720 TSecr=217604933

Screenshot of TCP Segment:



Screenshot of ACK:



Segment Num	Length of segment	Sequence Num	Time when segment sent	Time of ACK	Length of ACK
50	1516	56473	0.520801	0.520810	68
52	1516	57921	0.520821	0.520828	68
54	1516	59369	0.520843	0.520860	68
56	1516	60817	0.520866	0.520874	68
58	1516	62265	0.520889	0.520897	68

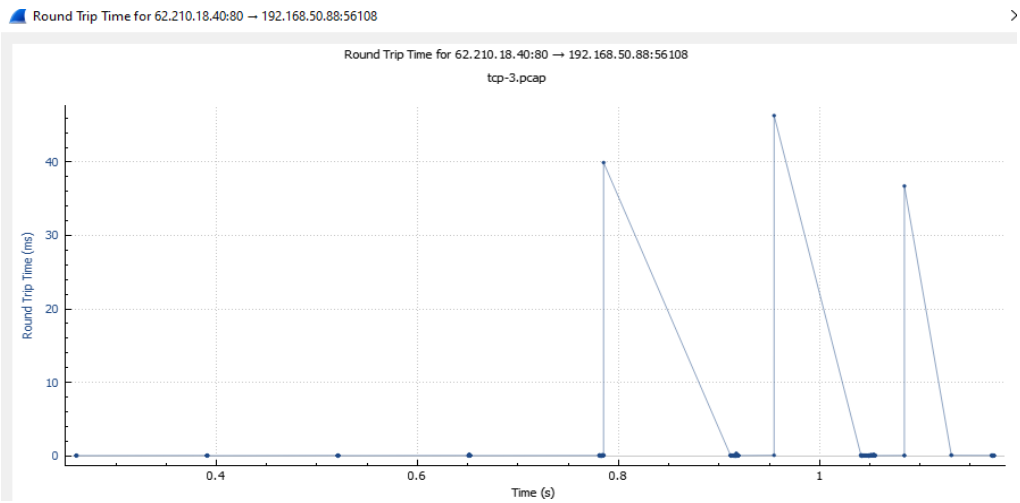
The maximum length is : **1516**

5.2. Given the difference between when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of these segments? What is the EstimatedRTT value 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 (From chapter 3 of the referred text book in the class) for all subsequent segments.
Place these calculated values appropriately in the table formed in 3.1 above.
EstimatedRTT = (1 − α) × EstimatedRTT + α × SampleRTT
where α = 0. 125 (that is, 1/8) [RFC 6298]

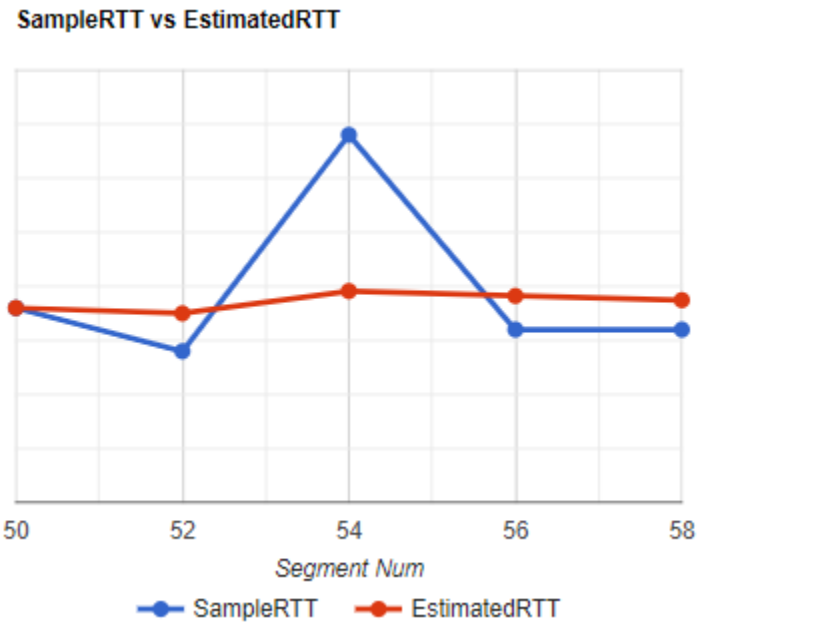
Segment Num	Sample RTT	Estimated RTT
50	0.000009	0.000009
52	0.000007	0.00000875
54	0.000017	0.000009781
56	0.000008	0.000009558
58	0.000008	0.000009363

5.3. Plot the RTT Graph for this TCP association, by picking any TCP packet out of the capture file, using the graph feature of Wireshark. Plot another graph manually from the table above (previous question) for Sample RTT and estimated RTT (Similar to "RTT samples and RTT estimates" graph from section "Round-Trip Time Estimation and Timeout" of the referred textbook in the class).

Solution:
Observe graph from Statistics → TCP Stream Graph → RoundTrip Time



Graph created manually using Segment Num as x-axis and Time as y-axis.



5.4. Comment on your understanding of Estimated RTT calculation and plotted RTT graphs.

Solution:
We find that RTT time is increasing or decreasing which may show increasing or decreasing congestion. This may be due to congestion in the network, that is the system may be busy with other tasks and as such not able to provide response immediately.

6. Calculate the overall throughput (bytes transferred per unit time) for this TCP conversation using sequence number and acknowledgement number information on the TCP header from the captured file. Explain how you calculated this value using Wireshark capture in this experiment along with relevant screenshots. Verify your

calculation with the one done by Wireshark using "Capture File properties". If you

observe the major difference in your calculation and one calculated by Wireshark, comment why and how ?

Solution:

<u>Throughput calculated manually:</u>

Throughput = (sequence num of last TCP segment - sequence num of 1st TCP segment) / (time of last ACK segment - time of 1st TCP segment)

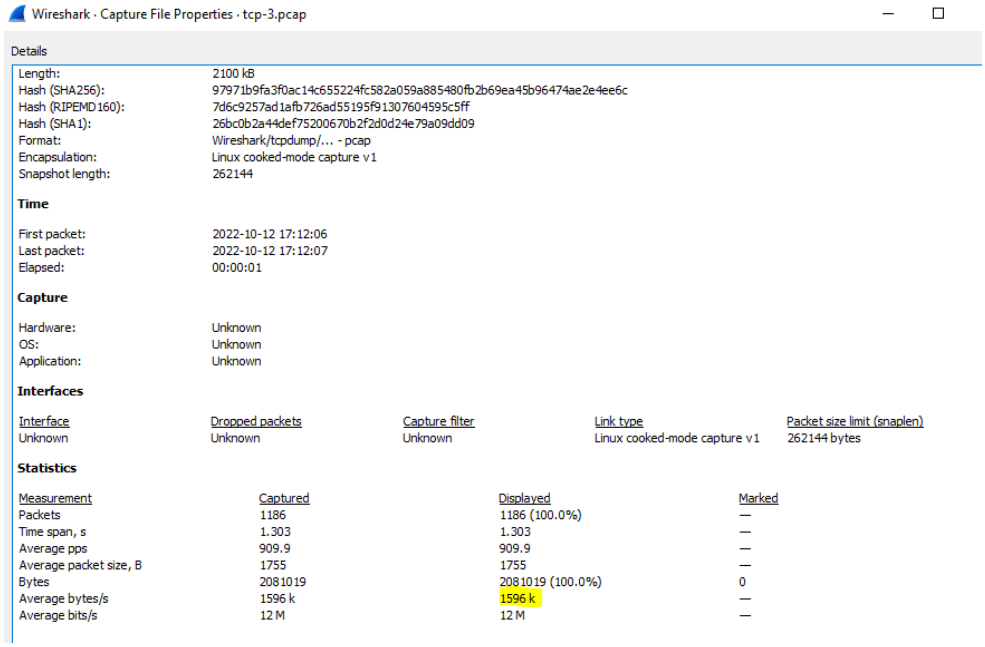
= (2000270-1)/(1.303419-0.260137)

= 1917285.0677 bytes/second

= 1872.34869893 KB/sec

<u>Throughput observed in the Capture File properties:</u>

We can see the throughput here is **1596KB/sec.**



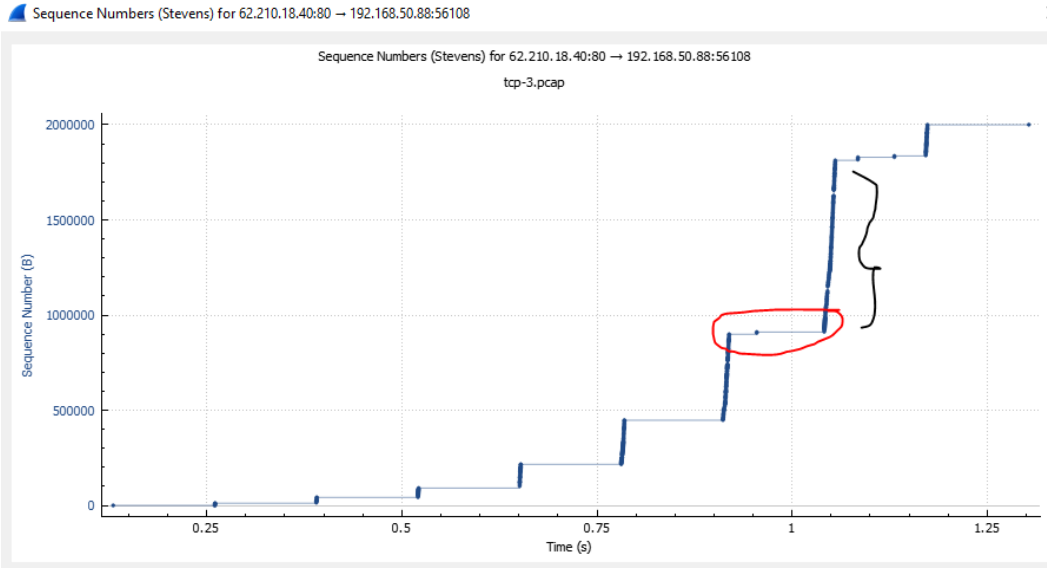
Comparing this to wireshark file properties, the reported value is close to the calculated value.

7. Using any active TCP segment (pick the packet of bulk data length, e.g: 7000+) involved in the download process from server to client, capture the TCP's functioning using the Time-Sequence-Graph (Stevens) plotting tool to view the sequence number versus time plot of segments being sent from the server to the client. Can you identify where TCP's slow start phase begins and ends, and where congestion avoidance takes over? If not possible, why ?

Solution:

Observe the graph under Statistics → TCP Stream graph → Time sequence(Stevens).

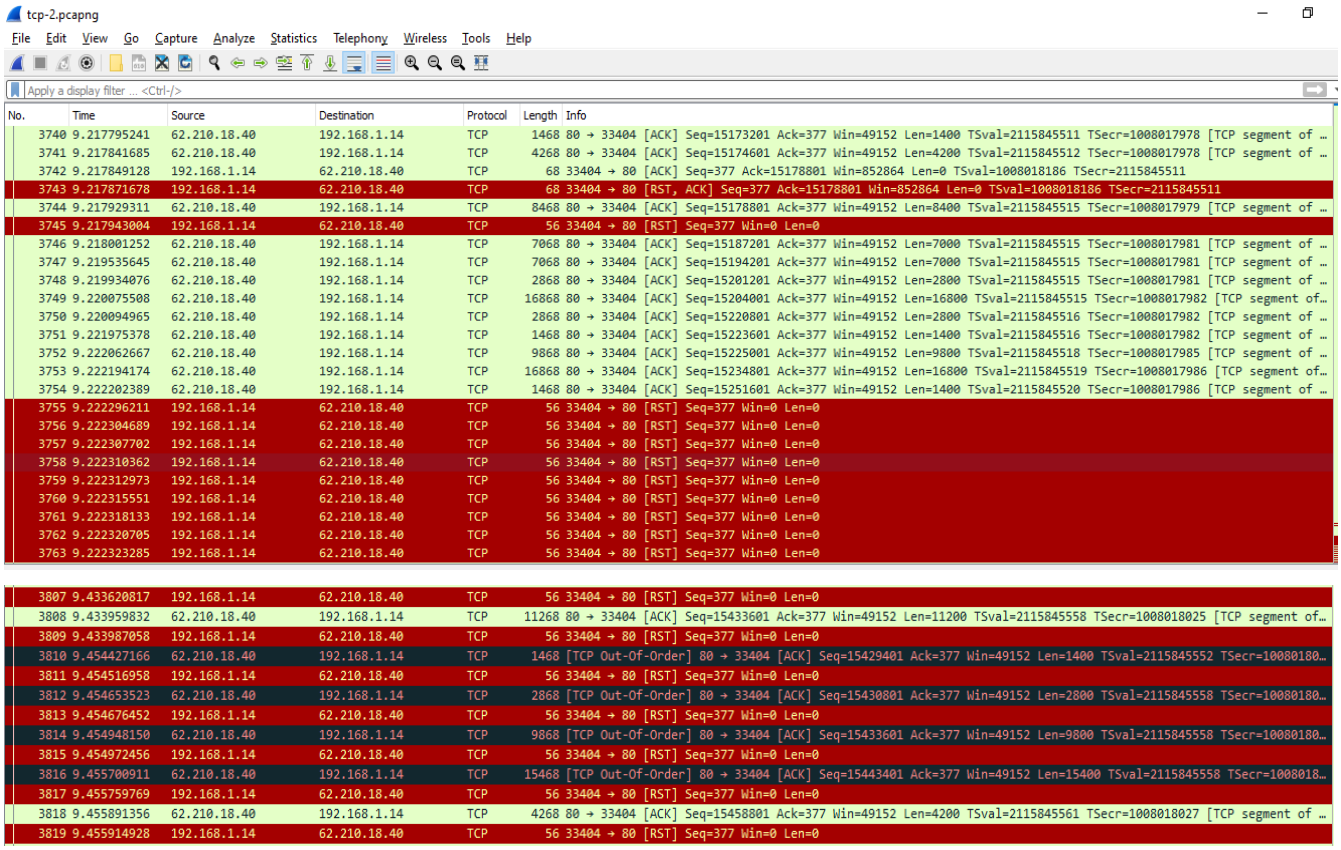
The part of the graph highlighted by red color denotes Slow start the congestion and the part of the graph marked with black color shows Multiplicative Increase and the places where distance increases rapidly can be considered as Congestion Control as segments are stopped to avoid congestion.



TASK 3

Use the attached task3-tcp2.pcap file to answer. Please use tcp as the display filter in the wireshark once you open it.
Observe and clearly explain with screenshots, how TCP connection gets terminated in this case, as well as which fields of TCP influence this.

Solution:



The TCP RST flag indicates that connection should be immediately terminated, and this happens mostly because of a fatal error, but the server keeps sending the packets and those packets are being lost . This is happening as there is no synchronization between them that's why we can observe the TCP out-of-order packets.

As no FIN flag is observed in any of the packets, the client keeps on sending RST packets trying to reset the connection.

PLAGIARISM STATEMENT

I certify that this assignment/report is my own work, based on my personal study and/or research and that I have acknowledged all material and sources used in its preparation, whether they be books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication. I also certify that this assignment/report has not previously been submitted for assessment in any other course, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that I have not copied in part or whole or otherwise plagiarized the work of other students and/or persons. I pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, I understand my responsibility to report honor violations by other students if I become aware of it.

Name of the student : SHRUSTI
Roll No : CS22MTECH11017