PART-A

Ans 1:

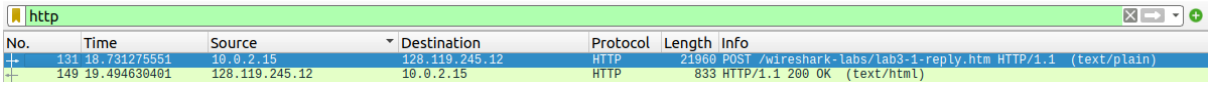
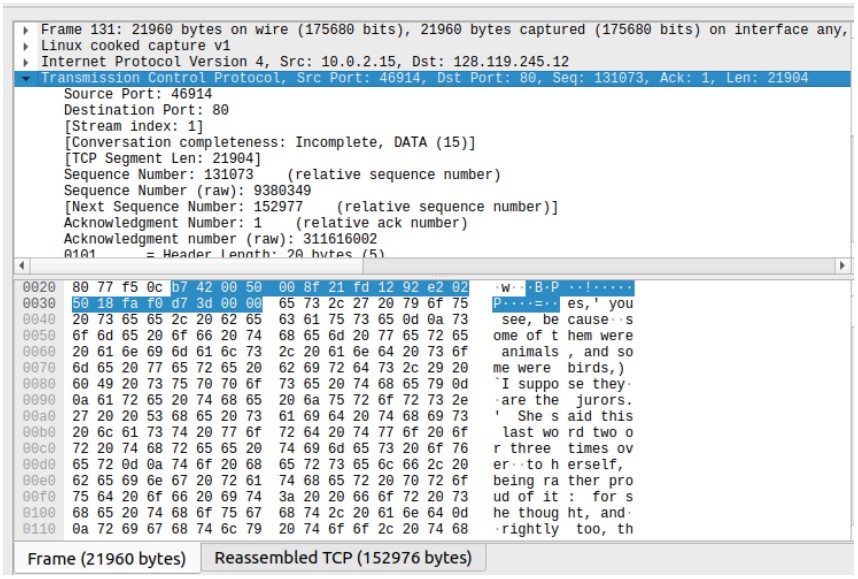


Figure 1: The HTTP POST message that uploaded alice.txt from your computer to gaia.cs.umass.edu



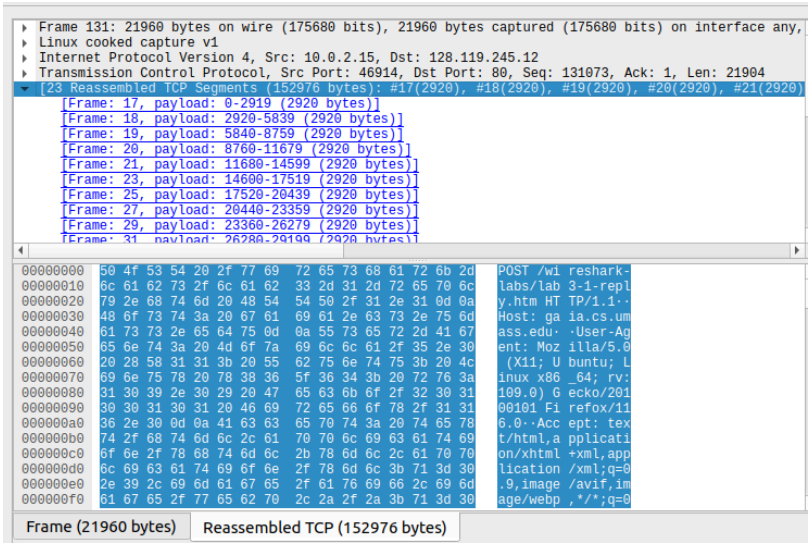


Figure 2, 3: expanding the HTTP POST message that uploaded alice.txt from your computer to gaia.cs.umass.edu

Here IP address of client computer (source) is **10.0.2.15**
And Source Port is **46914**

Ans 2:
Here IP address of the gain.cs.umass.edu is **128.119.245.12**
And Destination Port is **80**

Ans 3:

Here we can see 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 is **3744992237**
Where we can see that the relative sequence number is 0.

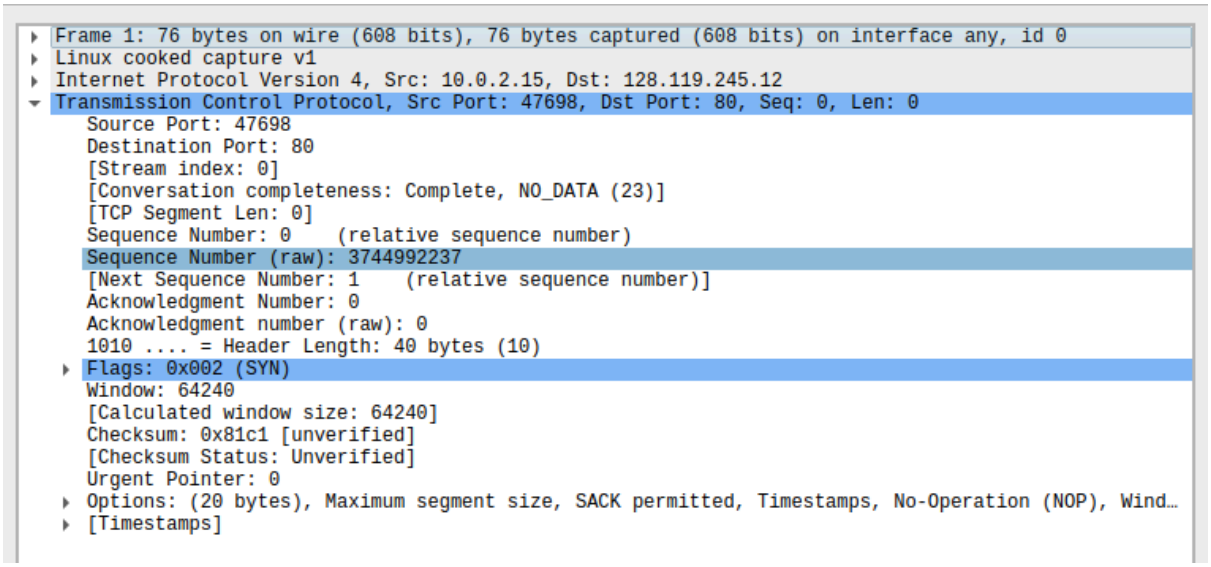


Figure 4: expanding the TCP SYN segment your computer to gaia.cs.umass.edu

Because of **Flags 0x002 (SYN)**

Here we can see the client computer (source) in SYN segment in options allow the permission of SACK

were gaia.cs.umass.edu in SYN segment have not given permission for the SACK

So we can not use selective Acknowledgments

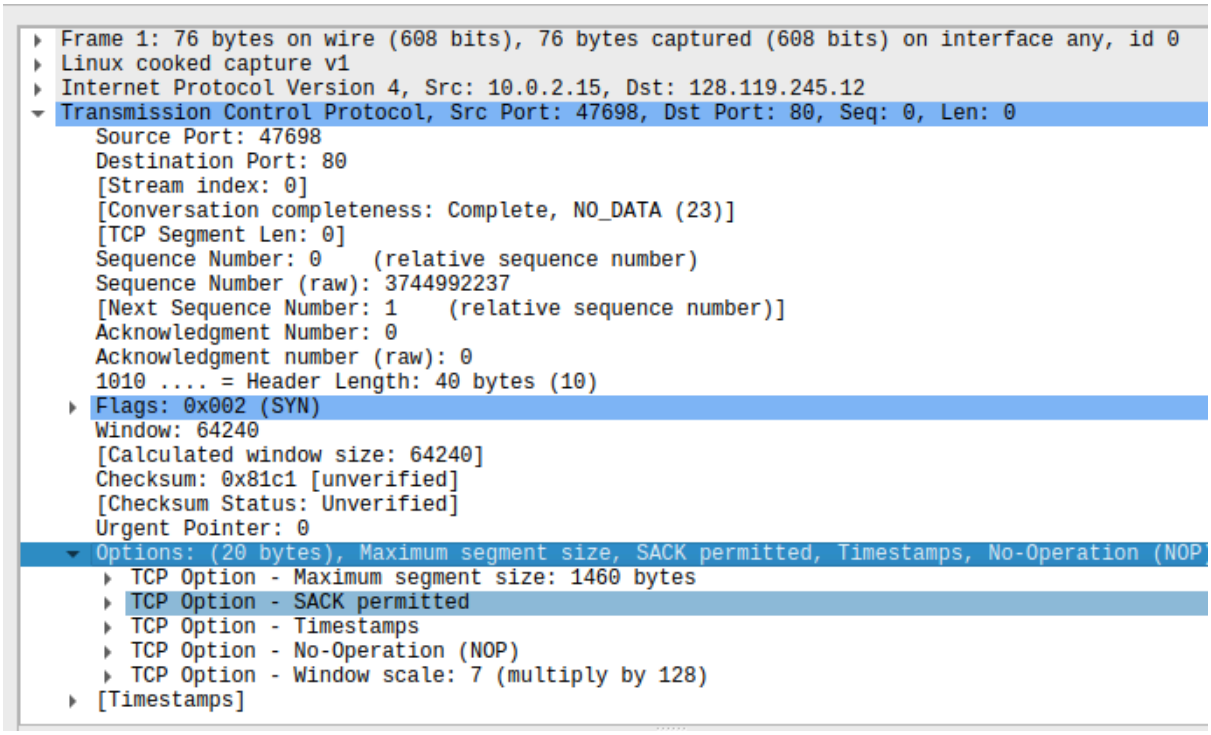


Figure 5: expanding the TCP SYN segment from your computer to gaia.cs.umass.edu

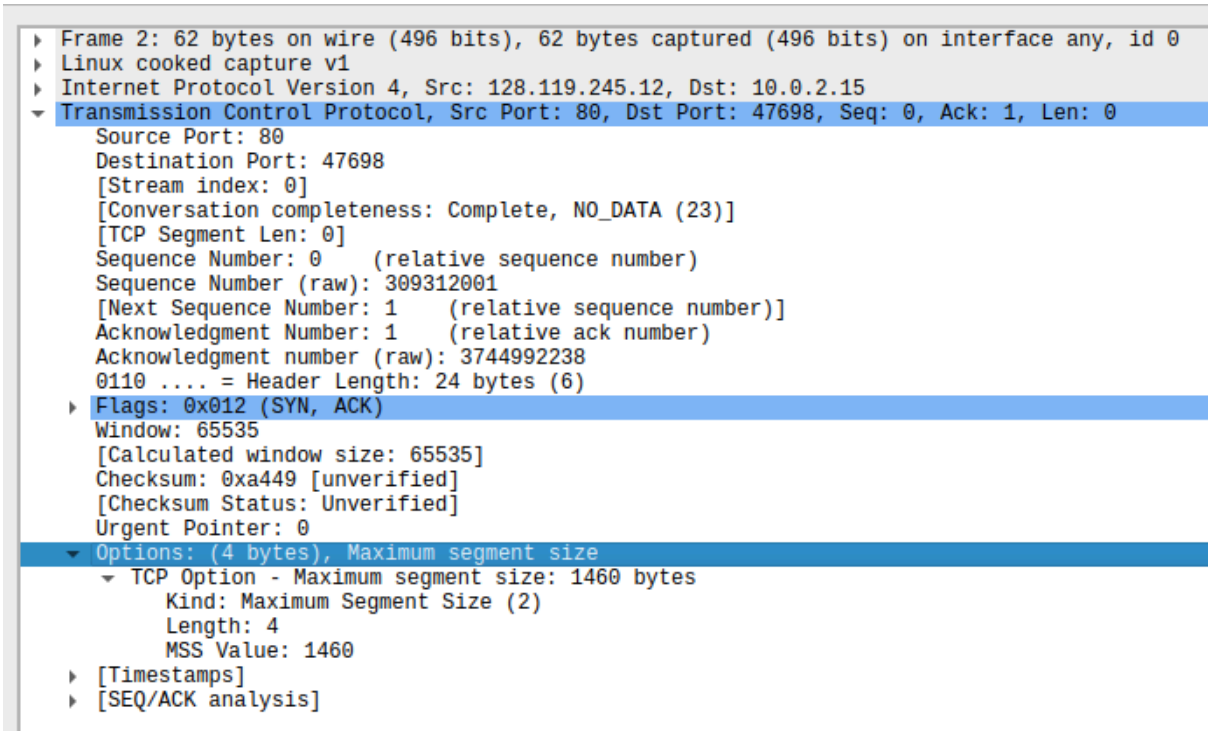


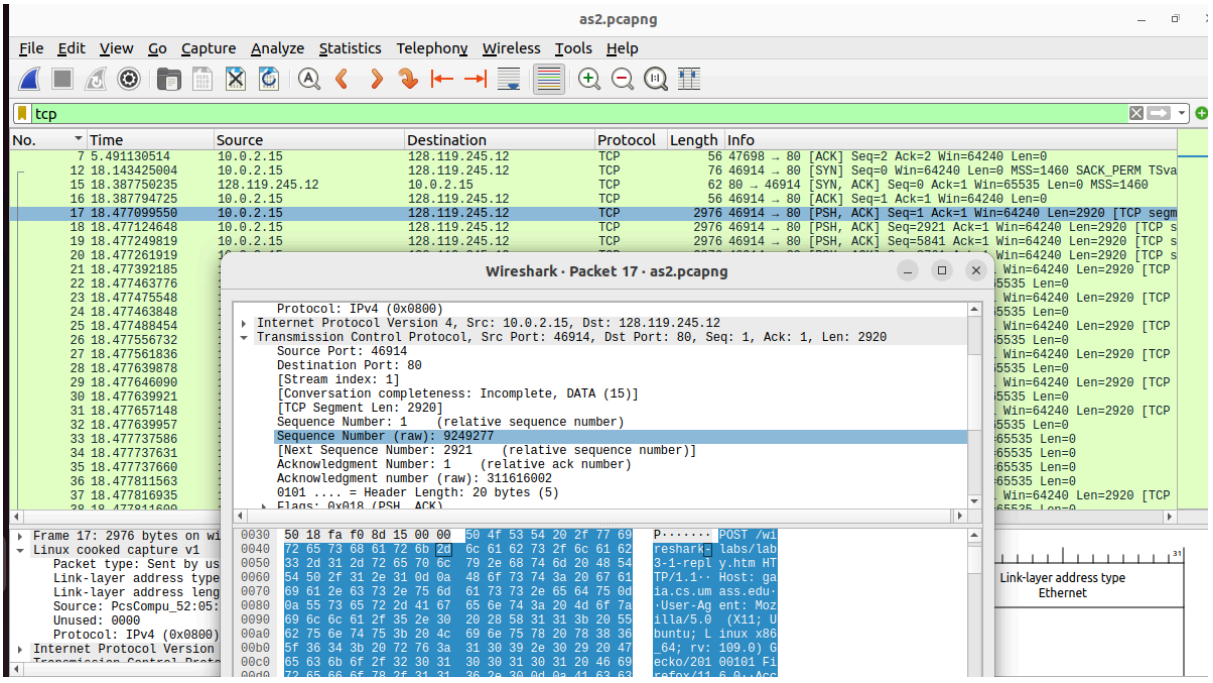
Figure 6: expanding the TCP SYNACK segment that to your computer from gaia.cs.umass.edu

Ans 4:
In figure 6 we can see sequence Number (raw) for asked segment is **309312001**

In figure 6 we can see Flags: 0x012 (SYN, ACK) it shows that the segment identifies as a SYNACK

in figure 6 we can see Acknowledgement number (raw) 3744992238 (which is 3744992237+1 where 3744992237 is sequence number of SYN segment as shown in figure 5 and 1 because length of the segment is 0 so we want next byte of the message, we can see the length of SYN segment in figure 5)

Ans 5:
Here in the given figure 7 we can see that data field contain the ascii 'POST'
Sequence number of this segment is **9249277**



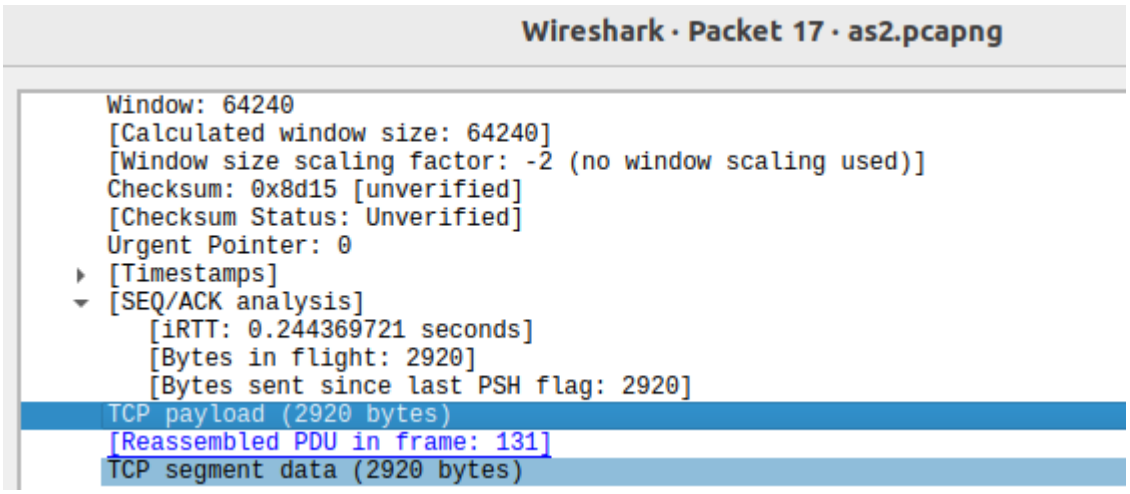
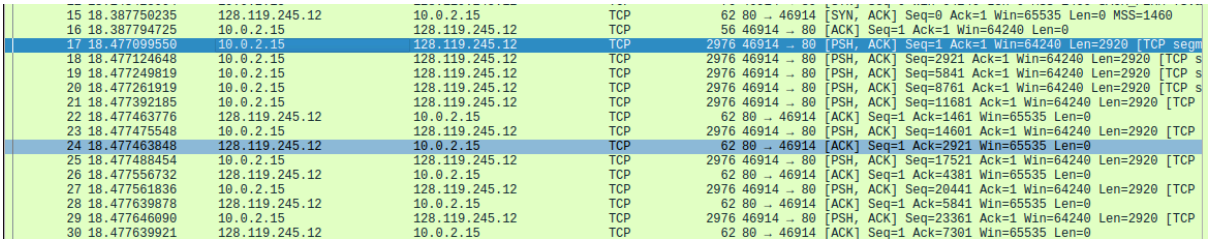


Figure 7 : expanding TCP segment containing the header of the HTTP POST command

Here 2920 bytes of data are contained in the payload (data) field of this TCP
No, all of the data in the transferred file alice.txt fit into this single segment

Ans 6:



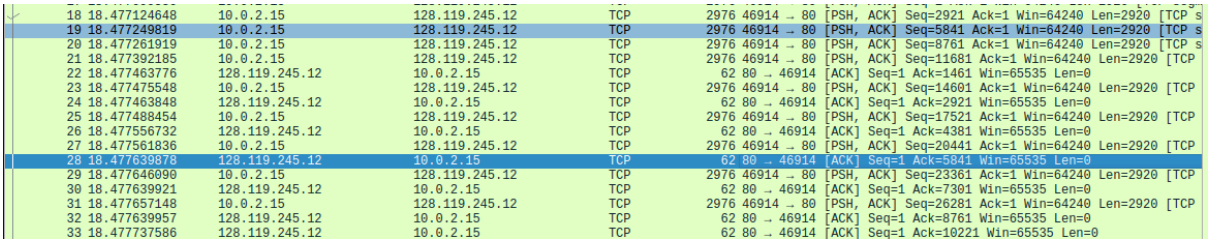
At what time was the first segment (the one containing the HTTP POST) in the data-transfer part of the TCP connection sent?

Ans: 18.477099550 sec.

At what time was the ACK for this first data-containing segment received?
What is the RTT for this first data-containing segment?

Ans: 18.477463848 sec. and RTT for this first data containing segment is **0.000364298 sec.**

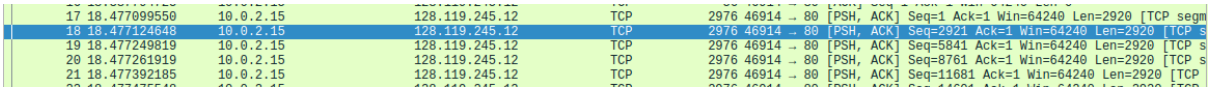
Ans: 18.477639878-18.477124648=0.0005130 sec.



Ans:
RTT for first packet : 18.477463848-18.477099550 = 0.000366
RTT for second packet : 18.477639878-18.477124648 = 0.000513

EstimatedRTT = (1-alfa) * Previously estimated RTT + alfa * Sample RTT
= (1-0.125) * 0.000366 + (0.125) * 0.000513
= 0.00032025 + 0.000064125 = **0.00038375**

Ans 7:

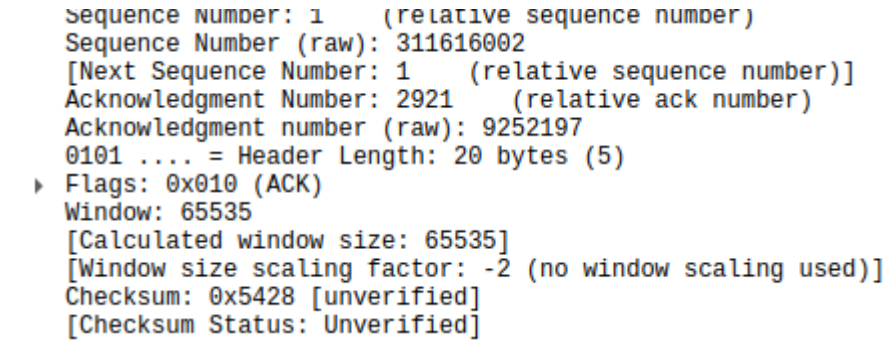


17 18.477099550	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=1 Ack=1 Win=64240 Len=2920 [TCP segm
18 18.477124648	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=2921 Ack=1 Win=64240 Len=2920 [TCP s
19 18.477249819	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=5841 Ack=1 Win=64240 Len=2920 [TCP s
20 18.477261919	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=8761 Ack=1 Win=64240 Len=2920 [TCP s
21 18.477392185	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=11681 Ack=1 Win=64240 Len=2920 [TCP

Here we can see 5 segments and their relative sequence number,
To calculate payload of the first 4 packets we can do relative sequence of 5th packet -1
= 11681-1
=11680
Now we add header length of each packets
=11680+80
=**11760**

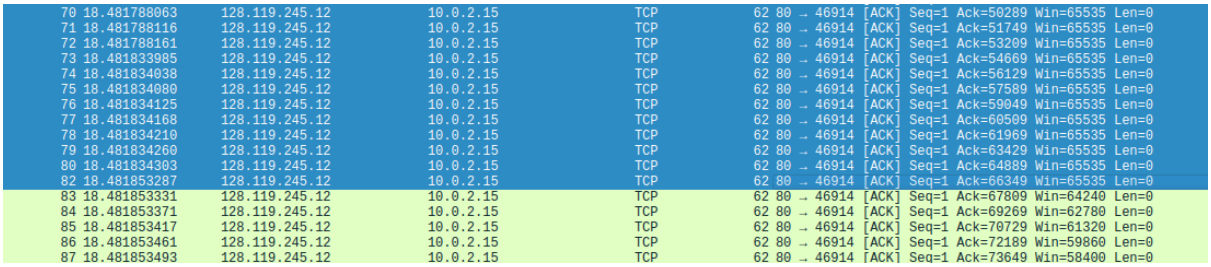
Or we can calculate other way is : = 4(2920+20) because each packet have payload=2920 bytes and header length is 20 bytes
=11760

Ans.8:

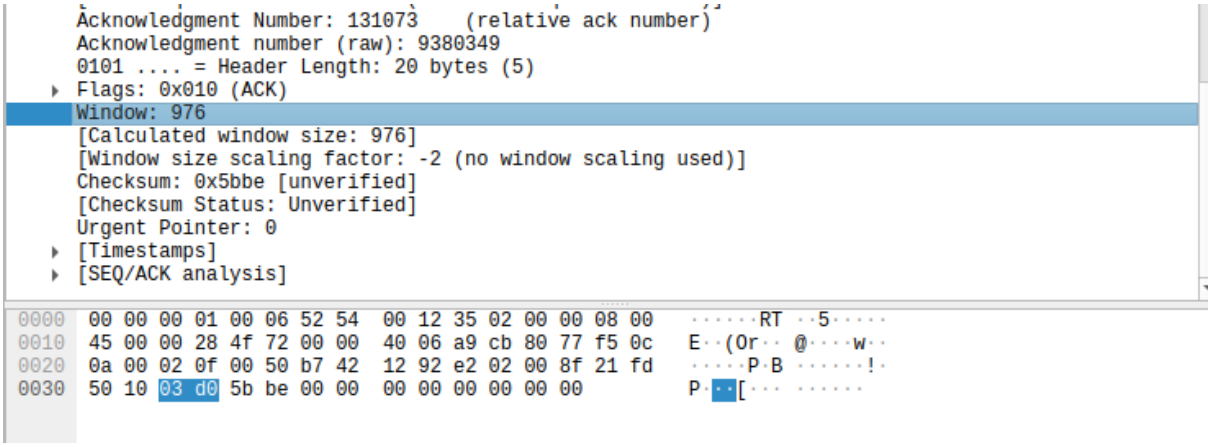


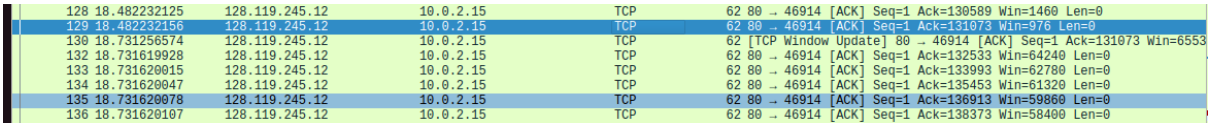
```
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 311616002
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 2921    (relative ack number)
Acknowledgment number (raw): 9252197
0101 .... = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
Window: 65535
[Calculated window size: 65535]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x5428 [unverified]
[Checksum Status: Unverified]
```

the minimum amount of available buffer space advertised to the client by gaia.cs.umass.edu among these first data-carrying TCP segments is 65535
And it is the same for all next five segments and there is no throttle. But after 45 packets we can see it started to decrease



70 18.481788063	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=50289 Win=65535 Len=0
71 18.481788116	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=51749 Win=65535 Len=0
72 18.481788161	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=53209 Win=65535 Len=0
73 18.481833985	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=54669 Win=65535 Len=0
74 18.481834038	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=56129 Win=65535 Len=0
75 18.481834080	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=57589 Win=65535 Len=0
76 18.481834125	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=59049 Win=65535 Len=0
77 18.481834168	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=60509 Win=65535 Len=0
78 18.481834210	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=61969 Win=65535 Len=0
79 18.481834260	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=63429 Win=65535 Len=0
80 18.481834303	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=64889 Win=65535 Len=0
82 18.481853287	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=66349 Win=65535 Len=0
83 18.481853331	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=67809 Win=64240 Len=0
84 18.481853371	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=69269 Win=62780 Len=0
85 18.481853417	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=70729 Win=61320 Len=0
86 18.481853461	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=72189 Win=59860 Len=0
87 18.481853493	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=73649 Win=58400 Len=0



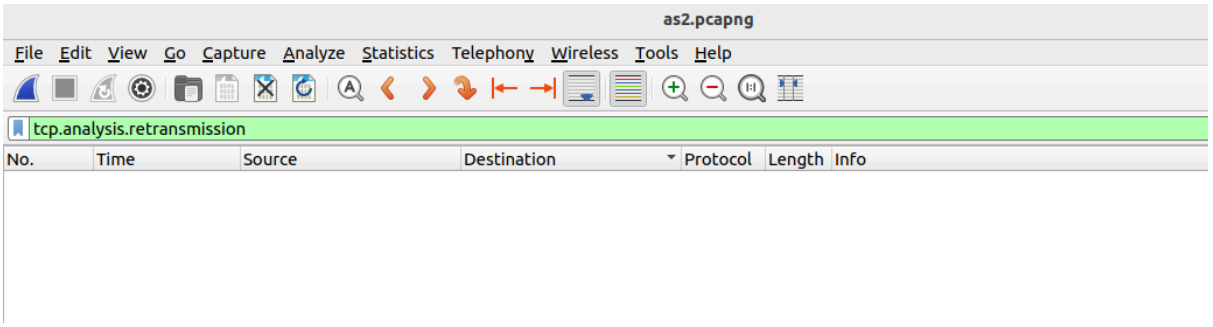


Here we see one of the the minimum amount of available buffer space advertised by the receiver to client is 976 and after it shows **TCP Window Update**

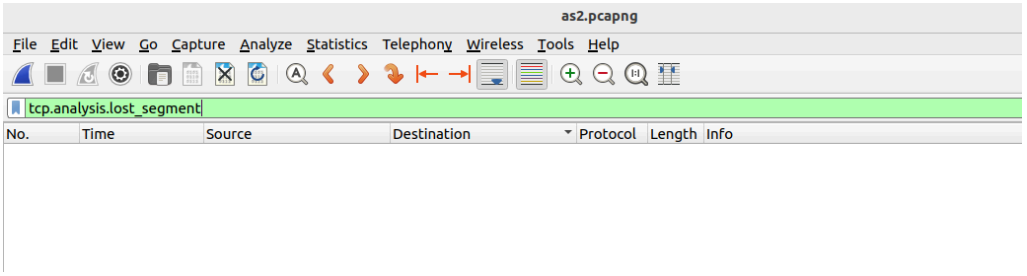
Ans 9:
the minimum amount of available buffer space advertised by the client to Gaia.cs.umass.edu is 64240 and it remains same for all the packets but in question 8 we can say that the window size of Gain.cs.umass.edu will not remain the same.

Ans 10:
Here we can see that there is no duplicate packets

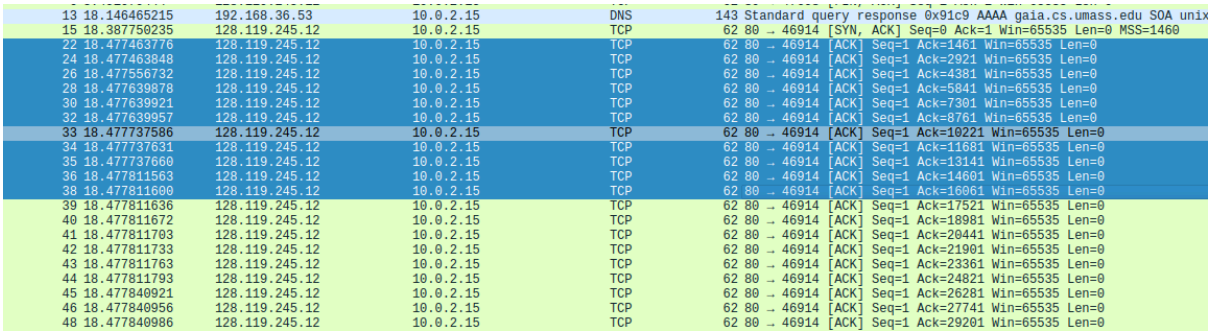
We can use tcp.analysis.retransmission to filter the packets



We can also use tcp.analysis.lost_segment to show any packers that lost and result in retransmission.



Ans 11:



13	18.146465215	192.168.36.53	10.0.2.15	DNS	143 Standard query response 0x91c9 AAAA gaia.cs.umass.edu SOA unix
15	18.387750235	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
22	18.477463776	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=1461 Win=65535 Len=0
24	18.477463848	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=2921 Win=65535 Len=0
26	18.477556732	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=4381 Win=65535 Len=0
28	18.477639878	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=5841 Win=65535 Len=0
30	18.477639921	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=7301 Win=65535 Len=0
32	18.477639957	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=8761 Win=65535 Len=0
33	18.477737586	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=10221 Win=65535 Len=0
34	18.477737631	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=11681 Win=65535 Len=0
35	18.477737660	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=13141 Win=65535 Len=0
36	18.477811563	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=14601 Win=65535 Len=0
38	18.477811600	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=16061 Win=65535 Len=0
39	18.477811636	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=17521 Win=65535 Len=0
40	18.477811672	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=18981 Win=65535 Len=0
41	18.477811703	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=20441 Win=65535 Len=0
42	18.477811733	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=21901 Win=65535 Len=0
43	18.477811763	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=23361 Win=65535 Len=0
44	18.477811793	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=24821 Win=65535 Len=0
45	18.477840921	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=26281 Win=65535 Len=0
46	18.477840956	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=27741 Win=65535 Len=0
48	18.477840986	128.119.245.12	10.0.2.15	TCP	62 80 → 46914 [ACK] Seq=1 Ack=29201 Win=65535 Len=0

1460 bytes of data does the receiver typically acknowledge in an ACK among the first ten data-carrying segments sent from the client to gaia.cs.umass.edu, I can't identify cases where the receiver is ACKing every other received segment among these first ten data-carrying segments.

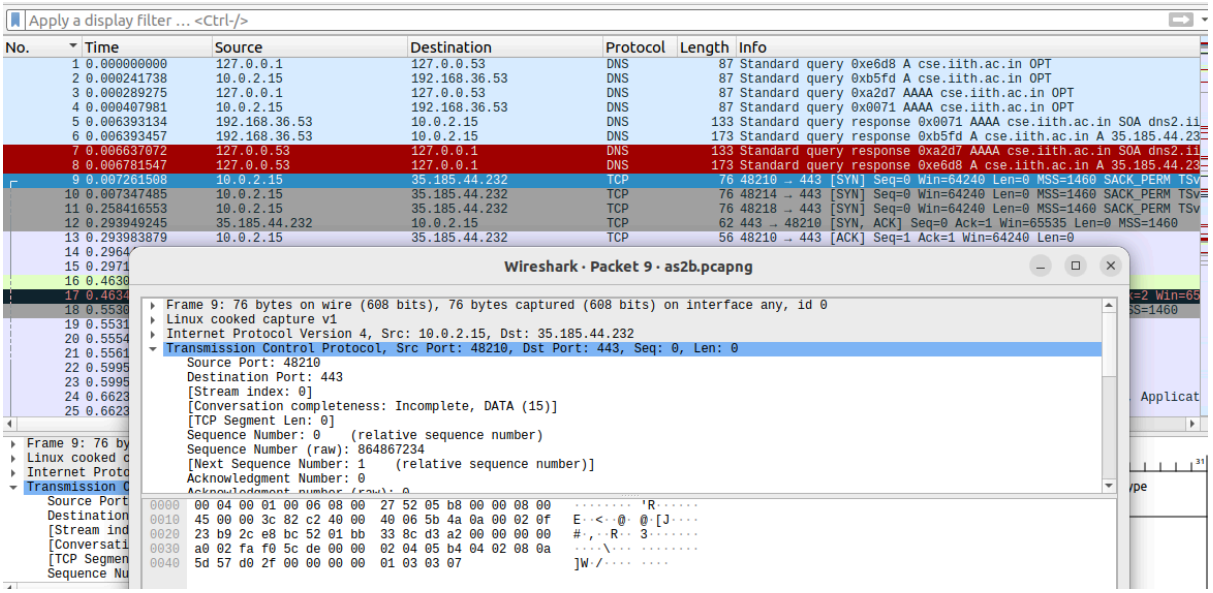
Ans 12:

Total data transferred = 152977-1 = 152976 (because 152977 is ack of last segment)
Know let time t is time difference between first packet send from client and last ack received at client = 18.731644940-18.477099550 = 0.254546390

Throughput = Total data transferred / time taken to transfer the data
= 152976/0.254546390
= 6009754.63
= 600 KBps

PART-B

Ans 1:



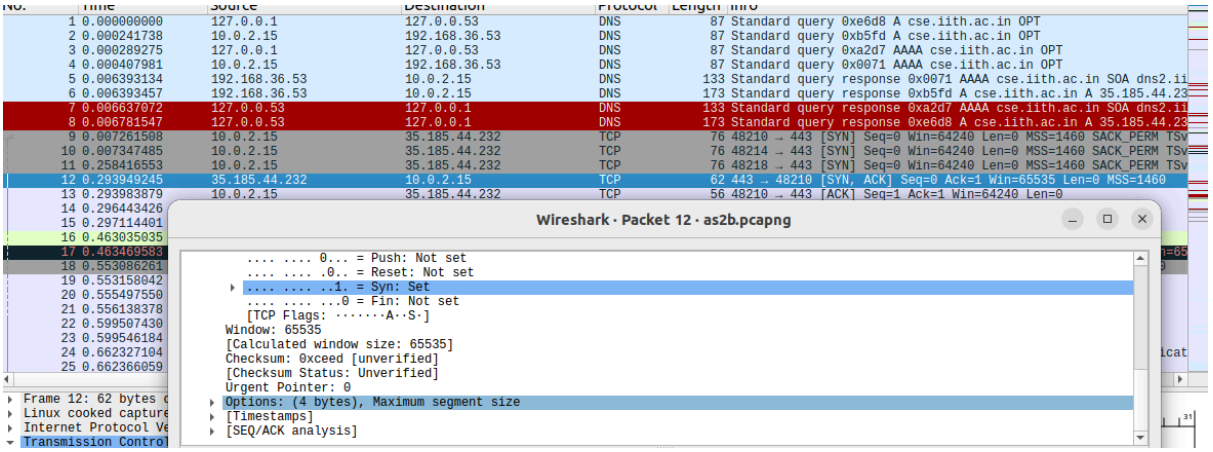
Here IP address of client computer (source) is **10.0.2.15**
And Source Port is **48210**

Ans 2:

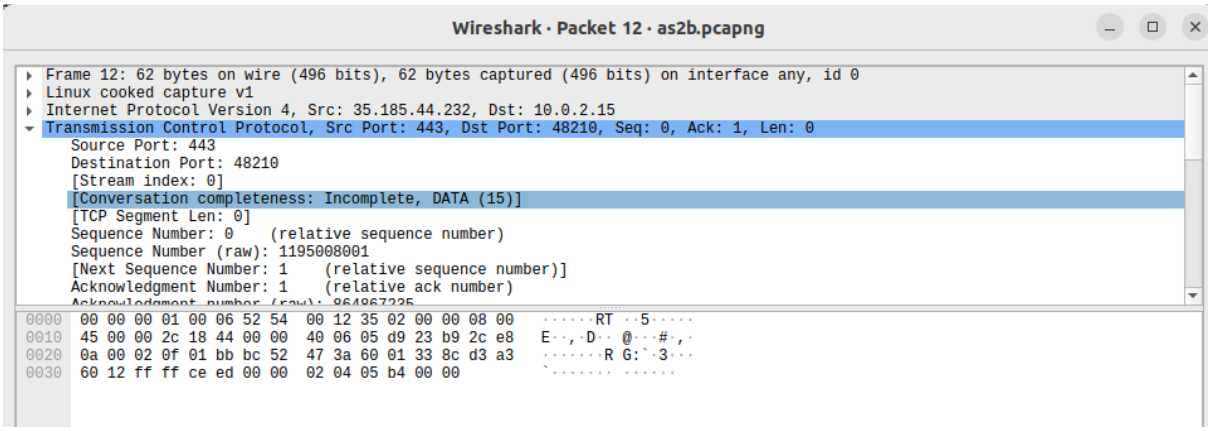
Here IP address of the https://cse.iith.ac.in/ is **35.185.44.232**
And Destination Port is **443**

Ans 3:
Here we can see the sequence Number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and https://cse.iith.ac.in/ is **864867234**
Where we can see that the relative sequence number is 0.

Because of control **Flags 0x002 (SYN)**



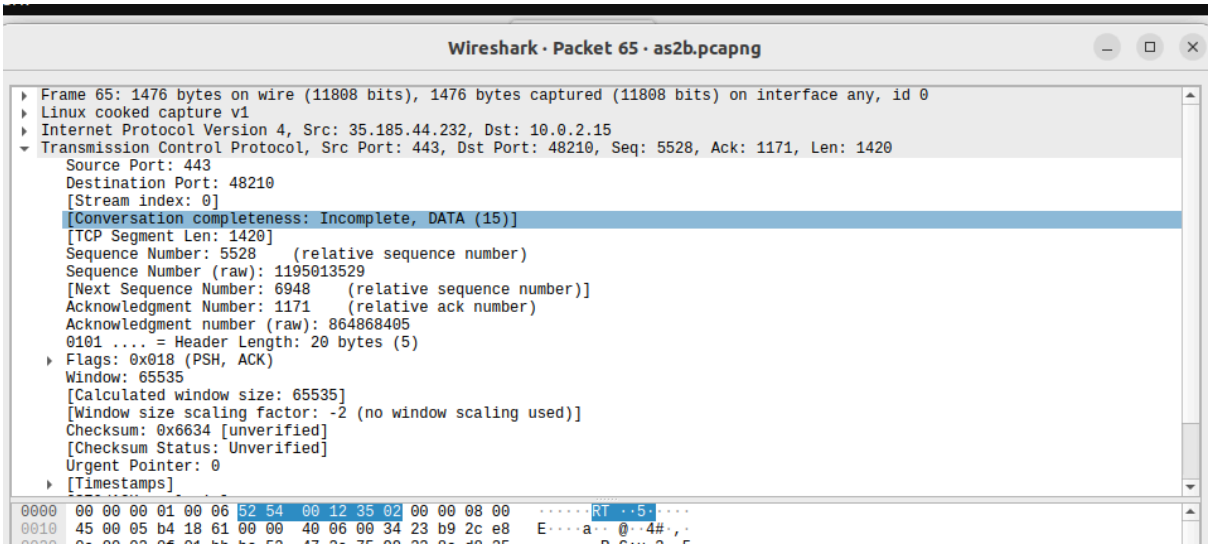
Here we can see the client computer (source) in SYN segment in options allow the permission of SACK
were https://cse.iith.ac.in/ in SYN segment have not given permission for the SACK
So we can not use selective Acknowledgments



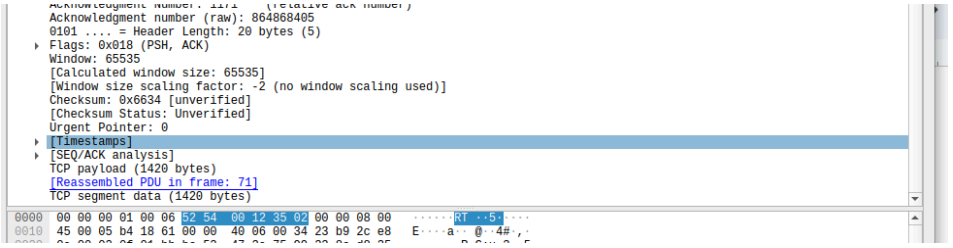
Ans 4:
In figure 6 we can see sequence Number (raw) for asked segment is **1195008001**
In above figure we can see Flags: 0x012 (SYN, ACK) it shows that the segment identifies as a SYNACK

in figure 6 we can see Acknowledgement number (raw) **864867234** (which is **864867233**+1 where **864867234** is sequence number of SYN segment as shown in figure and 1 because length of the segment is 0 so we want next byte of the message, we can see the length of SYN segment in figure)

Ans 5:

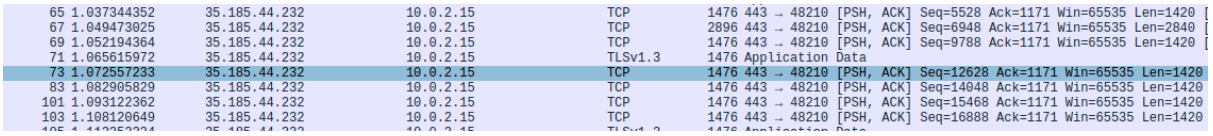


Here in the given Sequence number raw of this segment is **1195013529**

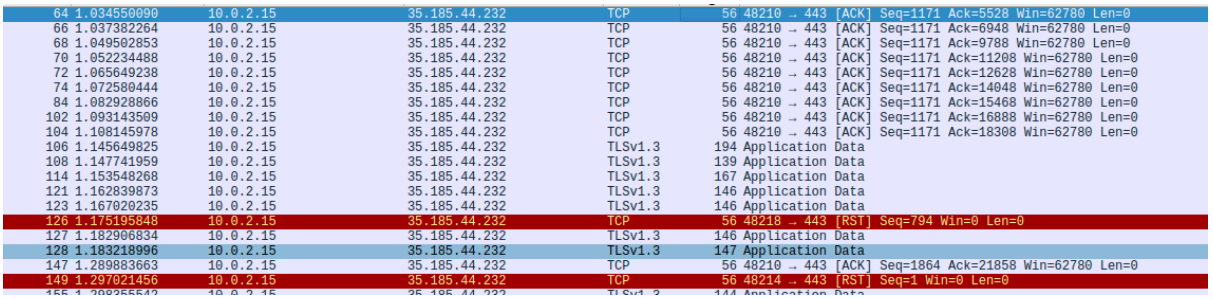


Here 1420 bytes of data are contained in the payload (data) field of this TCP
No, all of the data not transferred into this single segment

Ans 6:



-> **1.037344352 sec.**



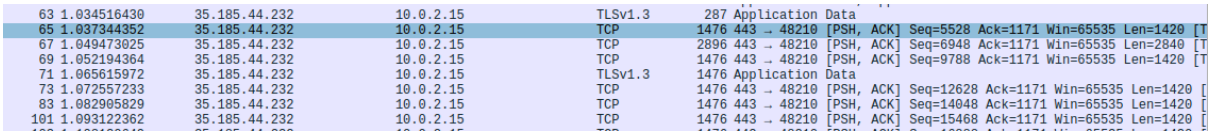
At what time was the ACK for this first data-containing segment received?
What is the RTT for this first data-containing segment?

Ack of the first segment is got at time **1.037382264 sec.** and RTT for this first data containing segment is **0.000038 sec.**
Rtt for the second segment will be **1.049502853-1.049473025 =0.000029 sec.**

RTT for first packet :**0.000038**
RTT for second packet : **0.000029**

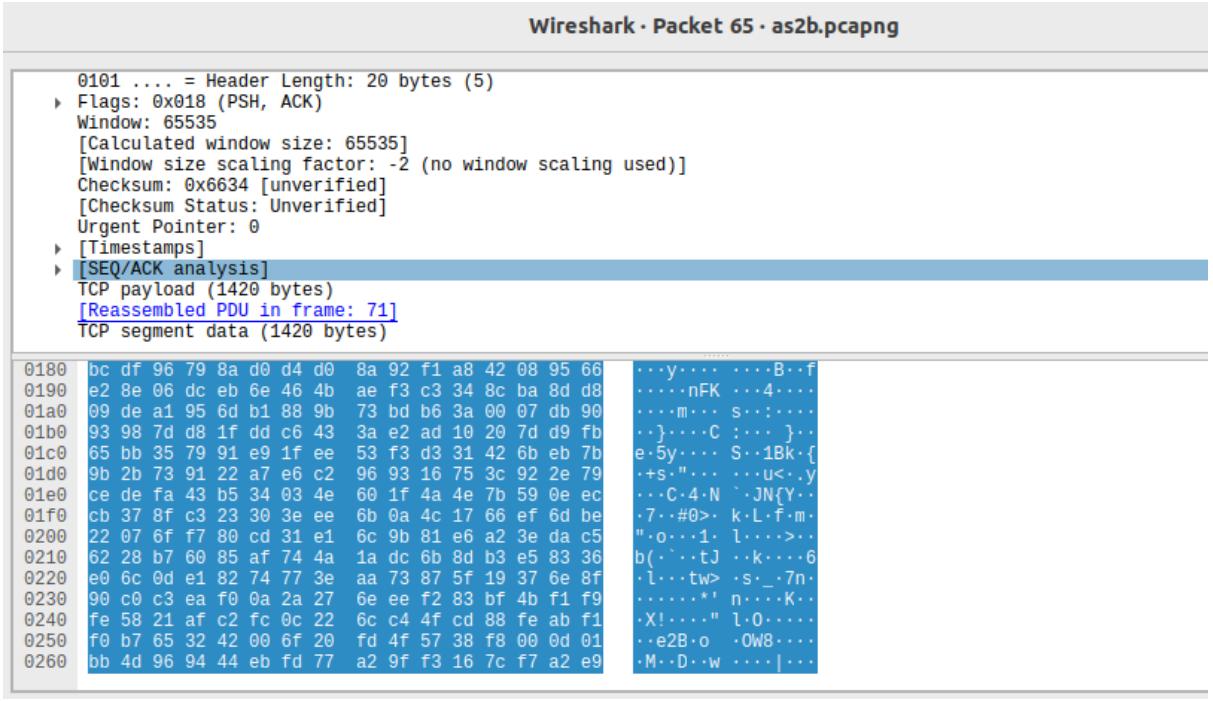
EstimatedRTT = (1-alfa) * Previously estimated RTT + alfa * Sample RTT
= (1-0.125) * **0.000038** + (0.125) * **0.000029**
= **0.000036875**

Ans 7:



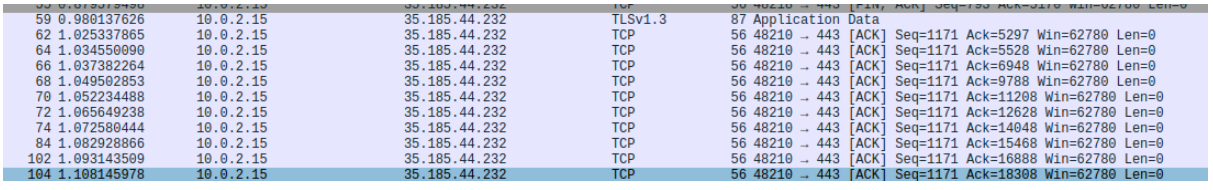
= 1420 + 20 + 2840 + 20 + 1420 + 20 + 1420 + 20
= 7200

Ans 8:



the minimum amount of available buffer space advertised by the client to https://cse.iith.ac.in/ among these first data-carrying TCP segments is 65535
And it is the same for all next five segments and there is no throttle.

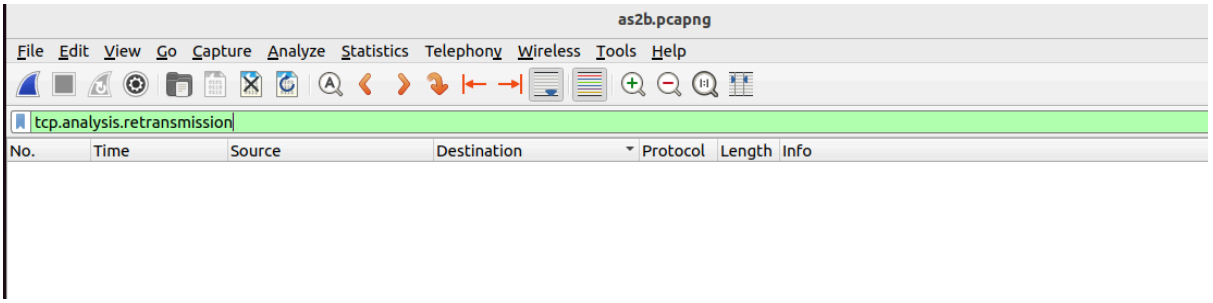
Ans 9:



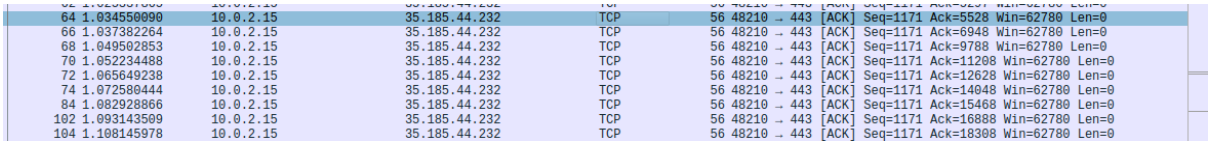
the minimum amount of available buffer space advertised to the client by https://cse.iith.ac.in/ is 62780 and it increase to 65535 but for question 8 for all the packets it is same as 65535

Ans 10:
Here we can see that there is no duplicate packets

We can use tcp.analysis.retransmission to filter the packets

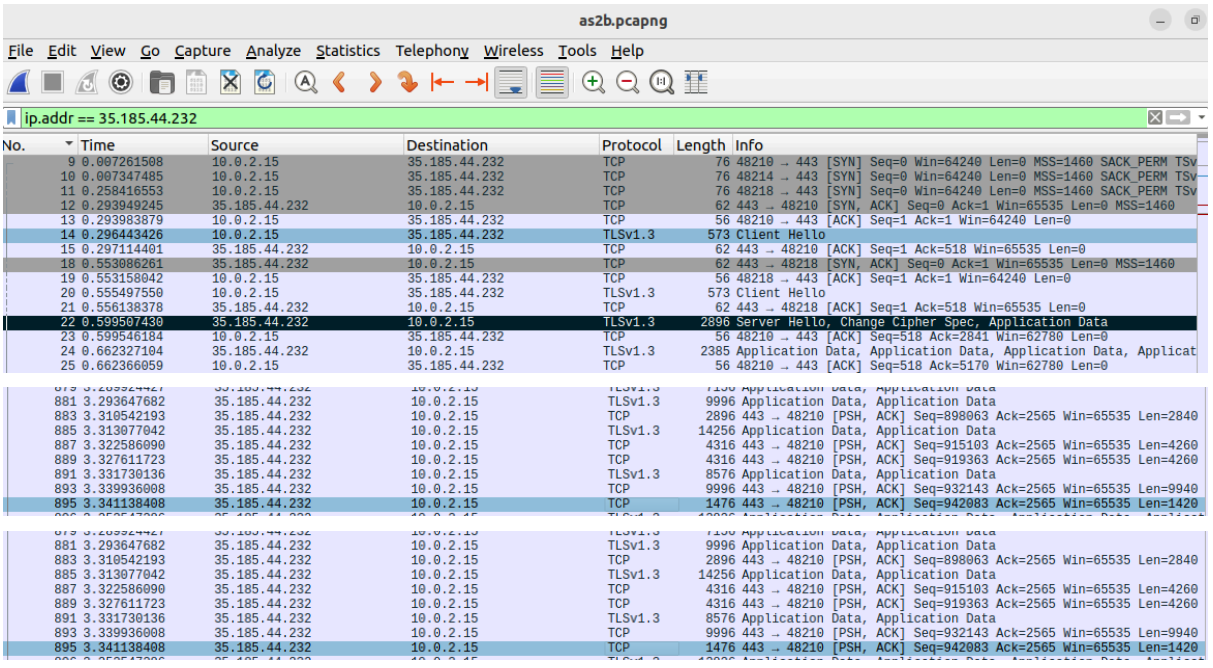


Ans 11:



1420 bytes of data does the receiver typically acknowledge in an ACK among the first ten data-carrying segments sent from the https://cse.iith.ac.in/ to client I can't identify cases where the receiver is ACKing every other received segment among these first ten data-carrying segments.

Ans 12:



Total data transferred = 942083-5170-1 = 936912 byte
Know let time t is time difference between first packet sent from client and last ack received at client = 3.341138408 - 0.007261= 3.33 sec.

Throughput = Total data transferred / time taken to transfer the data

= 936912/3.33
= 281354.95
= 281.3 KBps