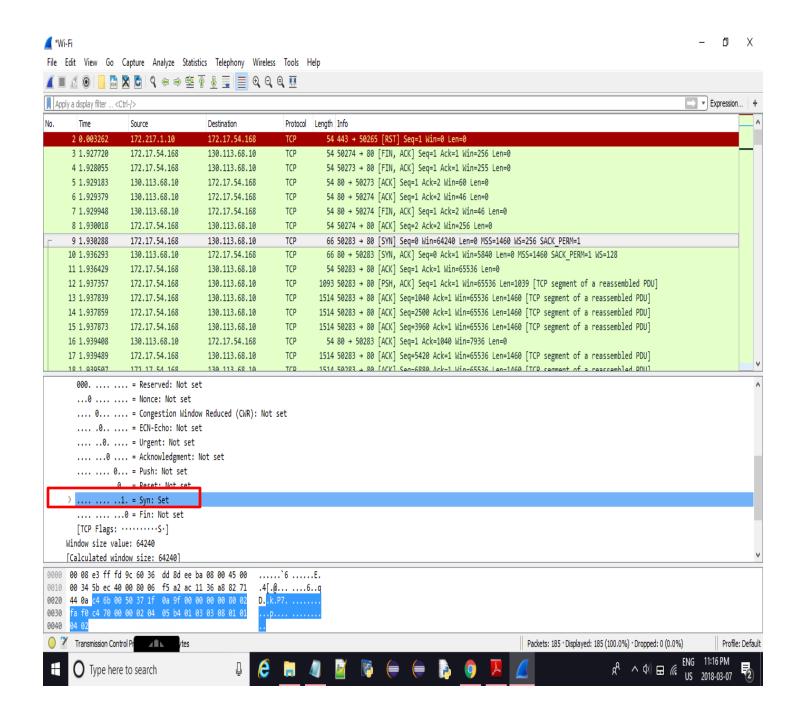
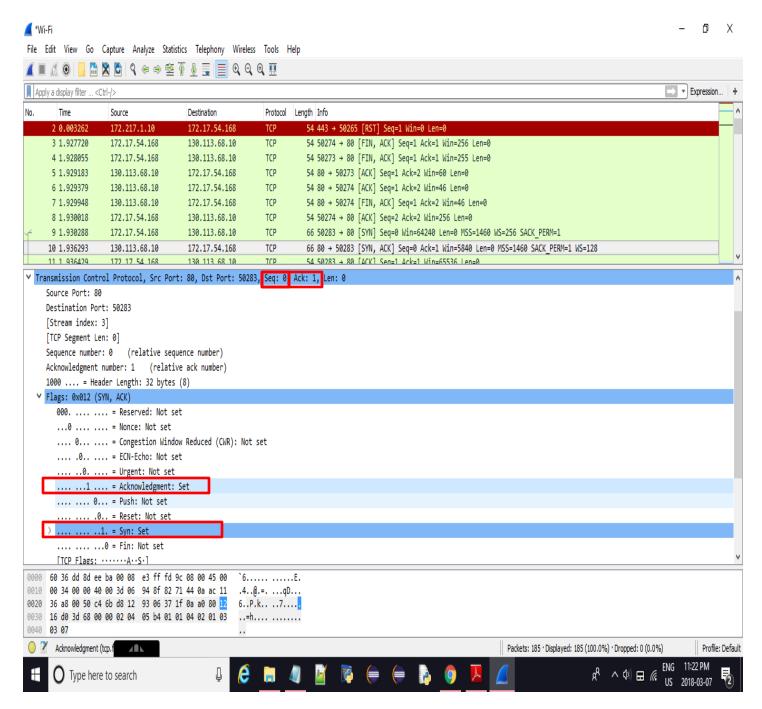


The Sequence number is 0.



The SYN flag is set to 1 which indicates that this segment is a SYN segment.

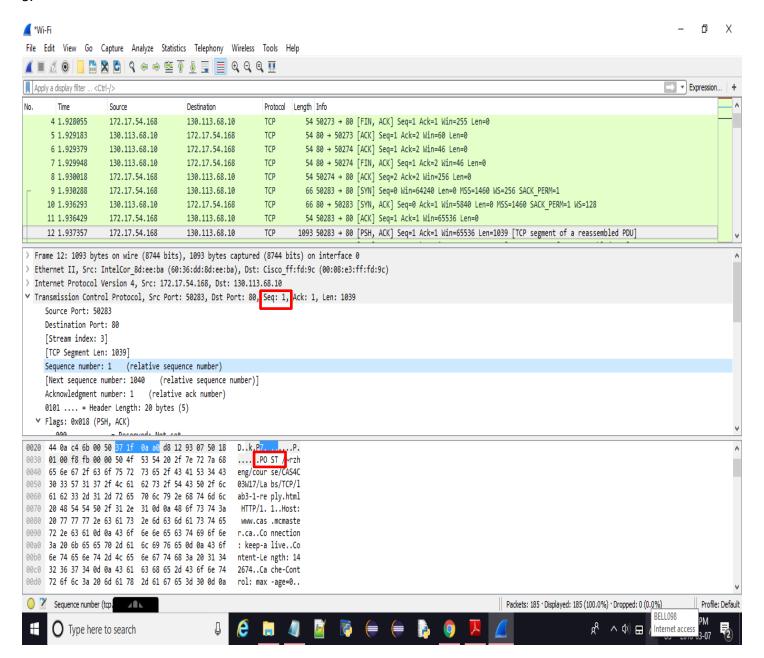


The sequence number is 0.

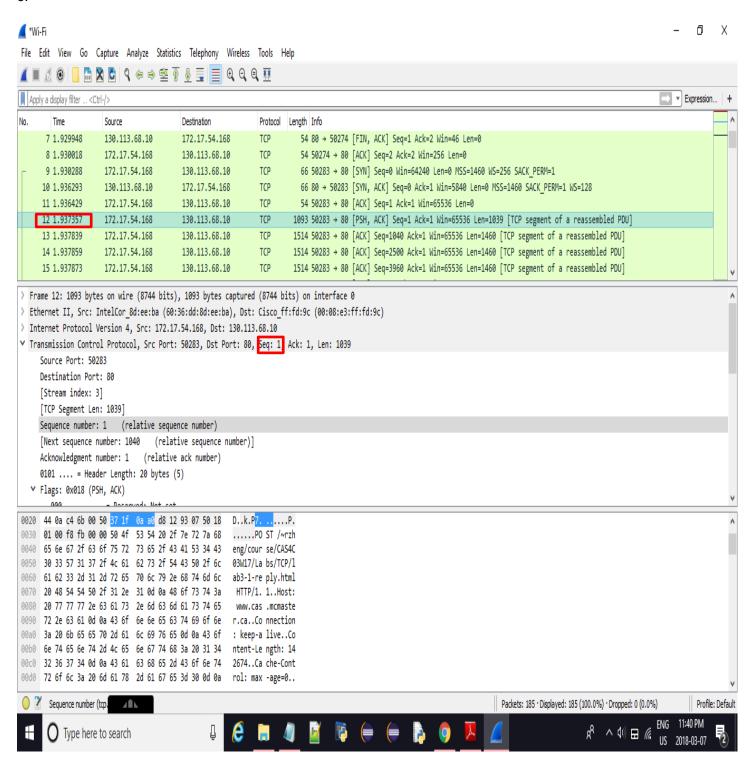
The value of Acknowledgement field is 1.

In this example, the initial sequence number of the SYN segment from the client computer is 0, and then the server adds 1 to the initial sequence number, therefore the value of the acknowledgement field in the SYN_ACK segment is 1.

A segment will be identified as a SYN_ACK segment if both SYN flag and Acknowledgement flag in the segment are set to 1.



The sequence number is 1.



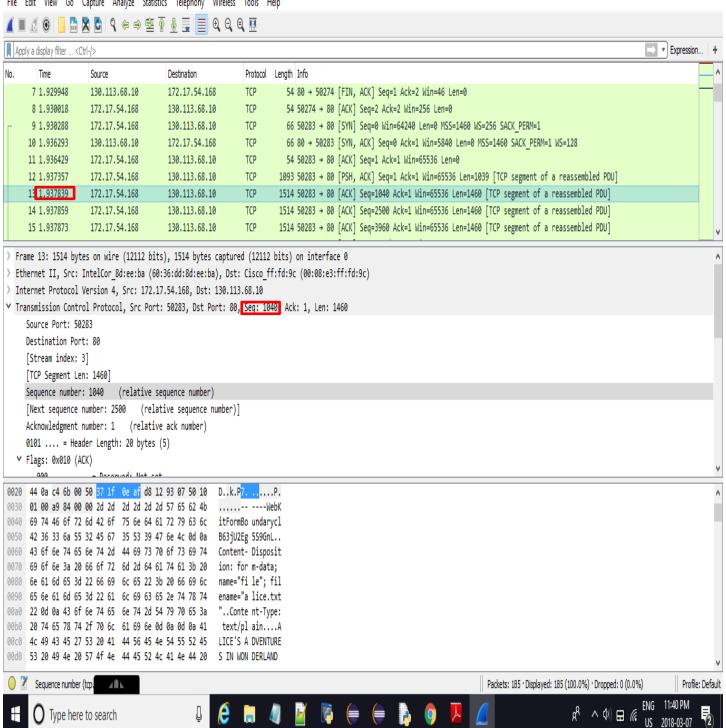
First Segment:

The Sequence number is 1.

It was sent at 1.937357 s.



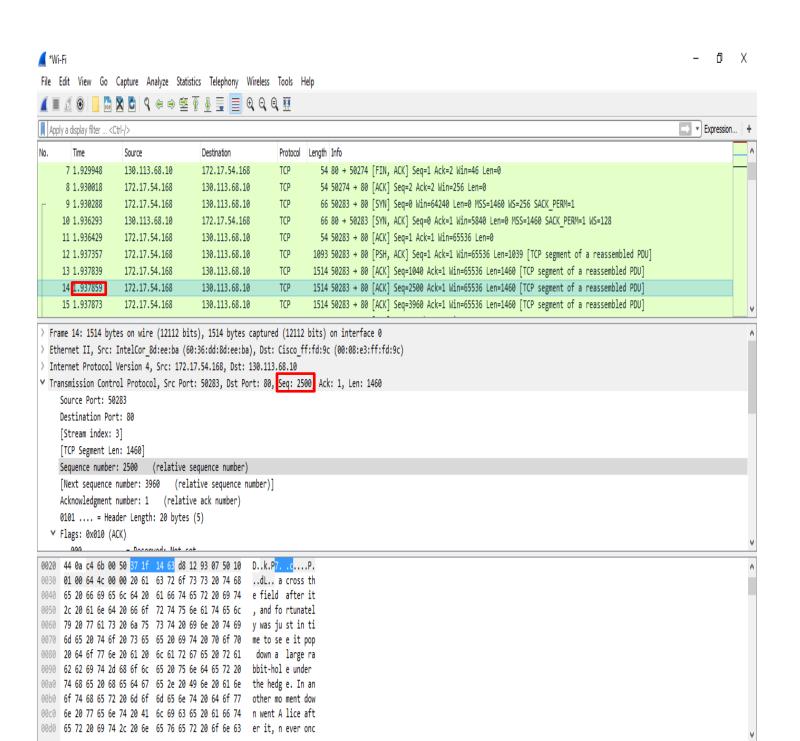
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help



Second Segment:

The Sequence number is 1040.

It was sent at 1.937839 s.



Packets: 185 · Displayed: 185 (100.0%) · Dropped: 0 (0.0%)

x⁸ ∧ 40) ⊞ (€

Profile: Default

ENG 11:40 PM

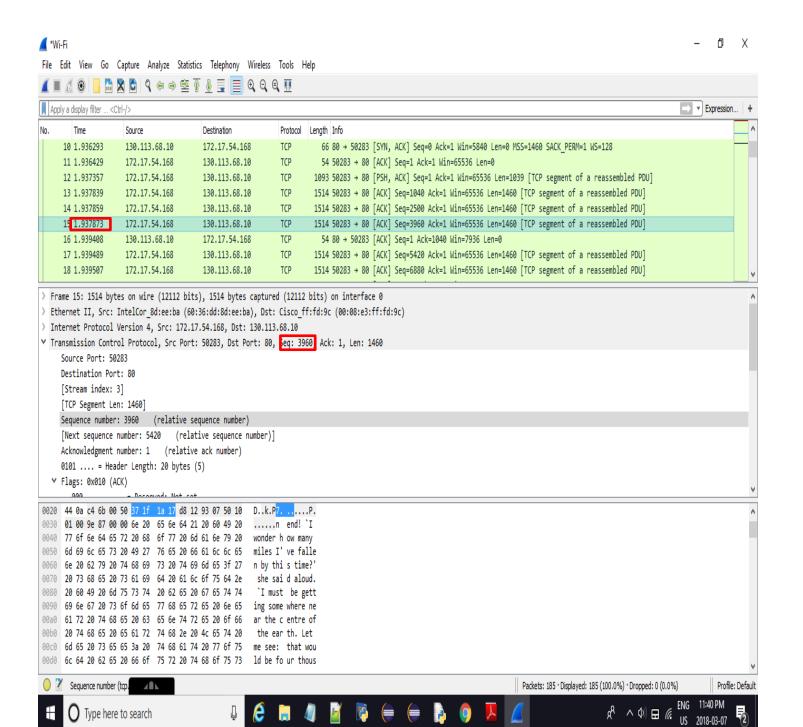
Third Segment:

Sequence number (tcp.

Type here to search

The Sequence number is 2500.

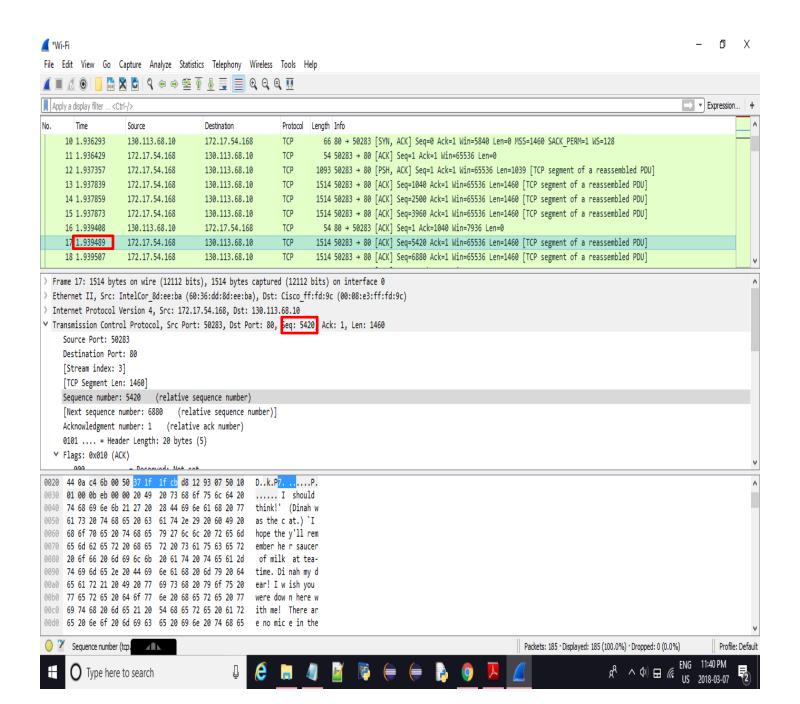
It was sent at 1.937859 s.



Fourth Segment:

The Sequence number is 3960.

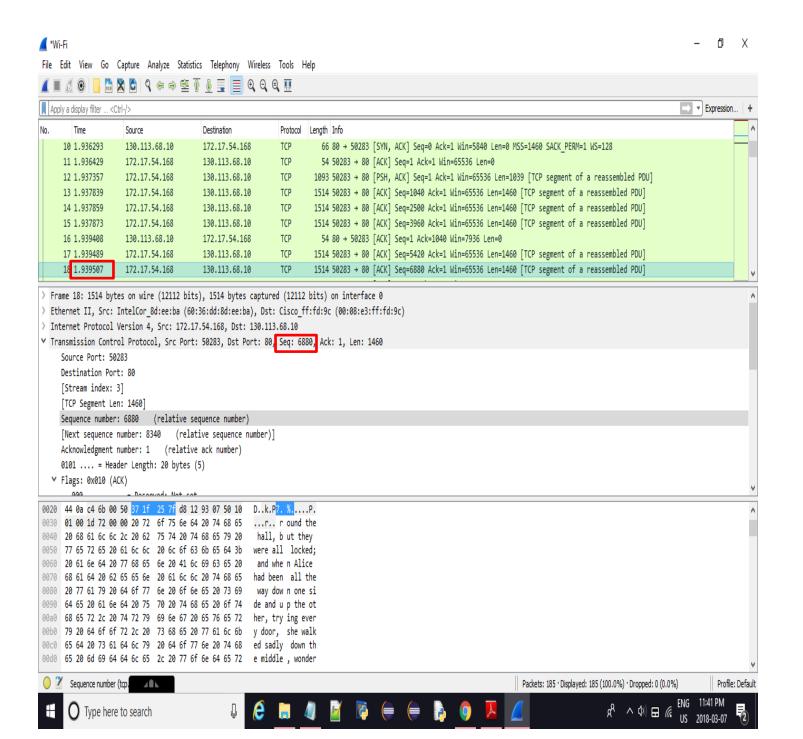
It was sent at 1.937873 s.



Fifth Segment:

The Sequence number is 5420.

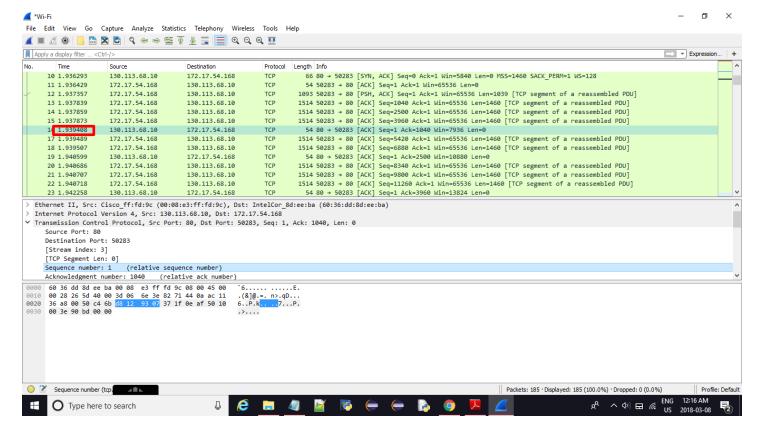
It was sent at 1.939489 s.



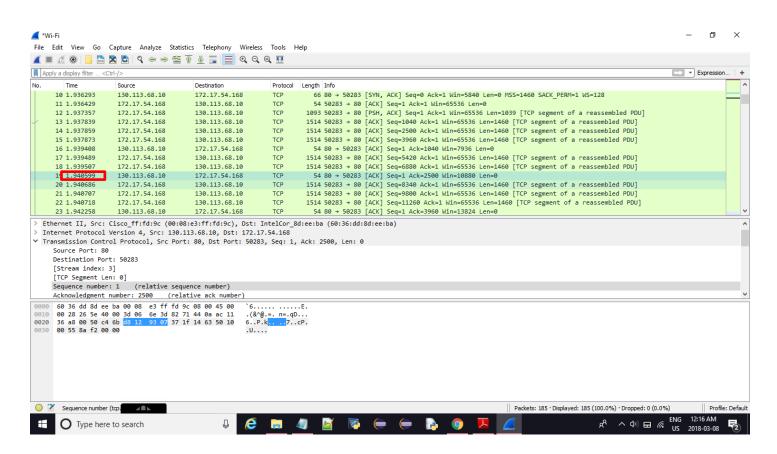
Sixth Segment:

The Sequence number is 6880.

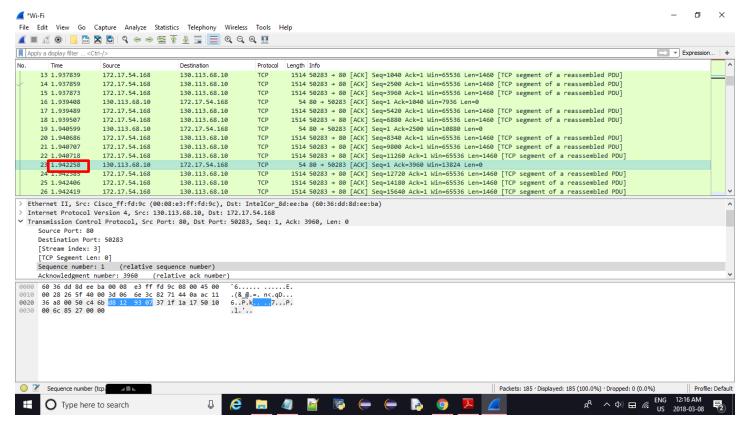
It was sent at 1.939507 s.



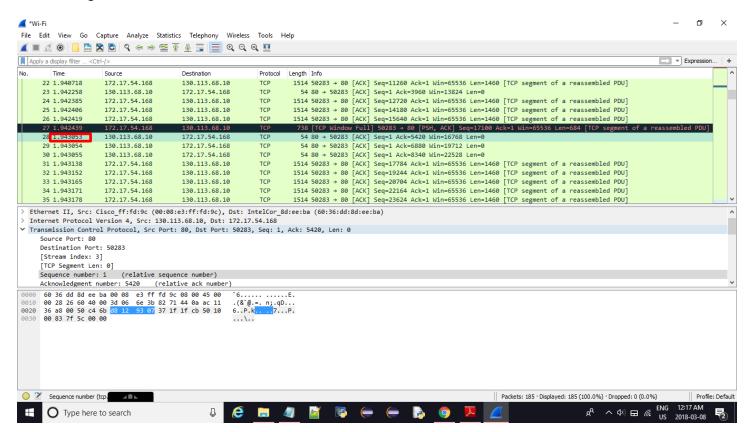
The first segment received at 1.939408 s.



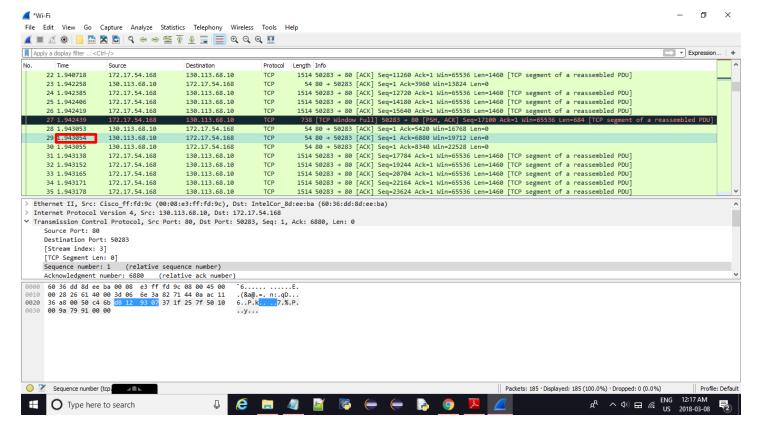
The second segment received at 1.940599 s.



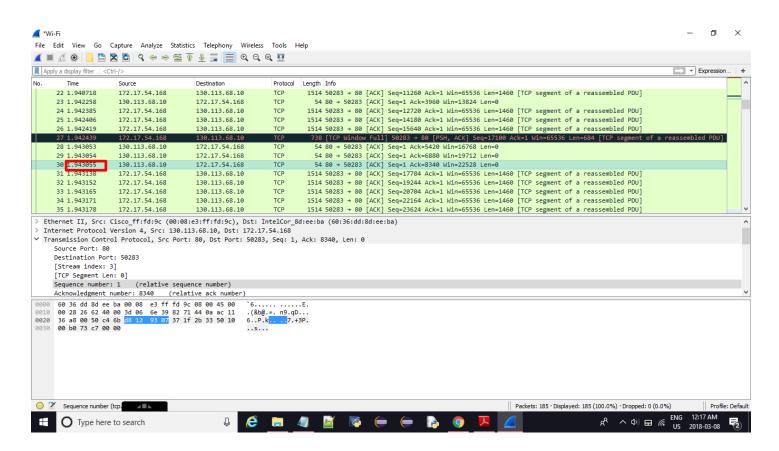
The third segment received at 1.942258 s.



The fourth segment received at 1.943053 s.



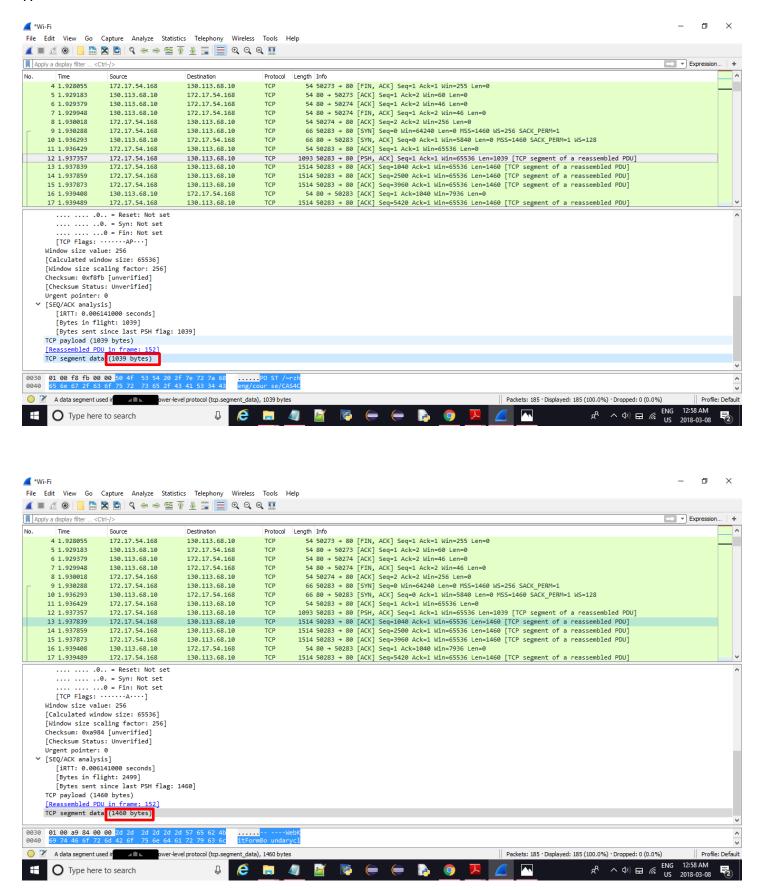
The fifth segment received at 1.943054 s.

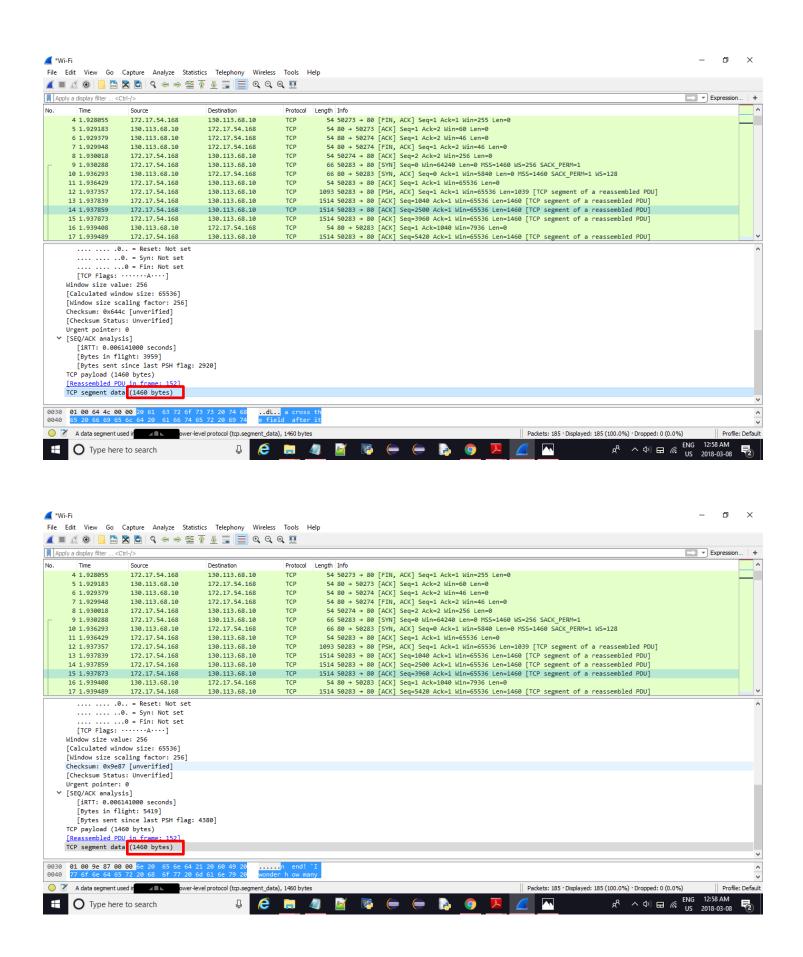


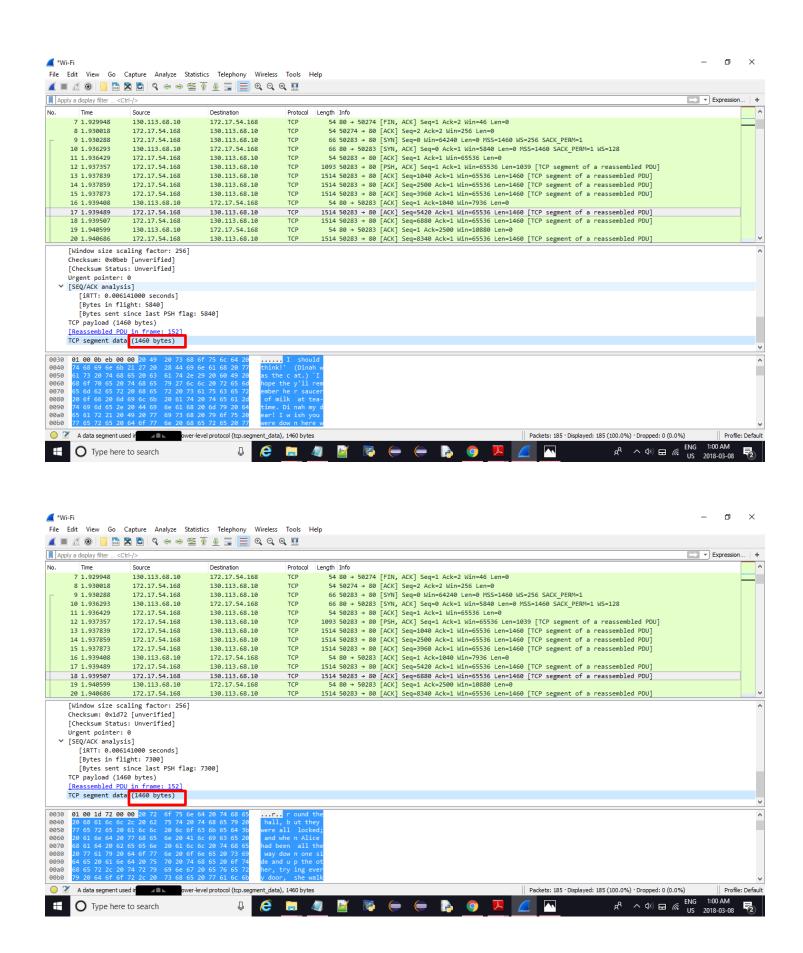
The sixth segment received at 1.943055 s.

Segment	Packet Number	Sequence Number	Time Sent(s)	Time ACK Received	Sample RTT (s) (Time received –
1	12	1	1.937357	1.939408	Time Sent) 0.002051
2	13	1040	1.937839	1.940599	0.00276
3	14	2500	1.937859	1.942258	0.004399
4	15	3960	1.937873	1.943053	0.00518
5	17	5420	1.939489	1.943054	0.003565
6	18	6880	1.939507	1.943055	0.003548

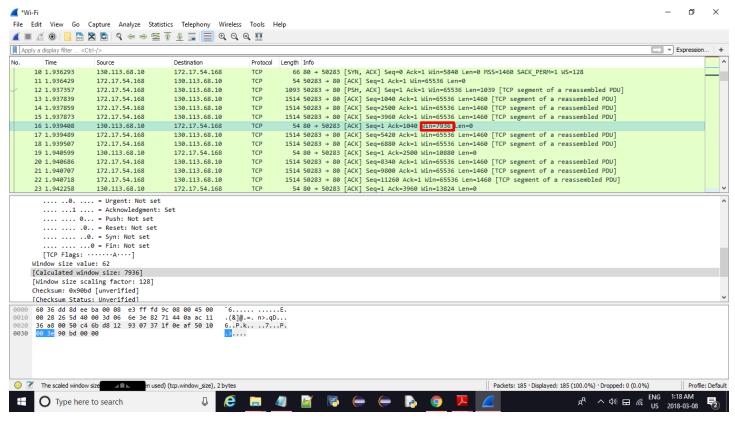
Segment	Estimated RTT			
	Estimated RTT = (1-alpha) * Estimated RTT + alpha * Sample RTT			
1	Estimated RTT = (1-0.125) * 0.002051 + 0.125 * 0.002051 = 0.002051			
2	Estimated RTT = (1-0.125) * 0.002051 + 0.125 * 0.00276 = 0.002139525			
3	Estimated RTT = (0.875) * 0.002139525 + 0.125 * 0.004399 = 0.002421718			
4	Estimated RTT = (0.875) * 0.002139525 + 0.125 * 0.00518 = 0.00251958			
5	Estimated RTT = (0.875) * 0.00251958 + 0.125 * 0.003565 = 0.00265025			
6	Estimated RTT = (0.875) * 0.00265025+ 0.125 * 0.003548 = 0.002762468			



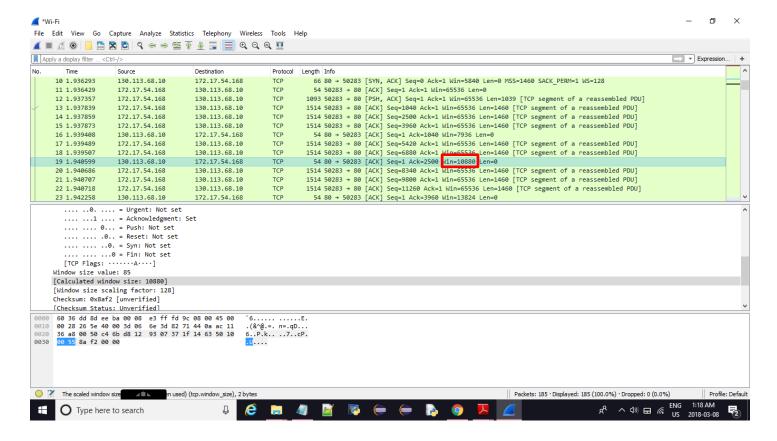




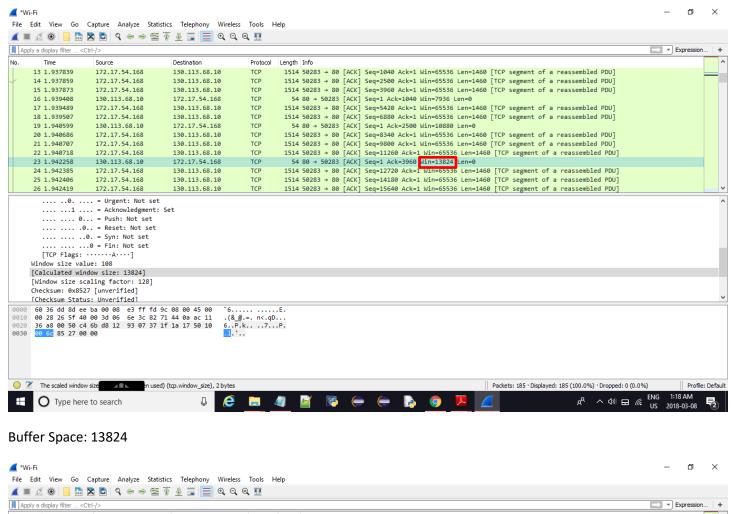
8.

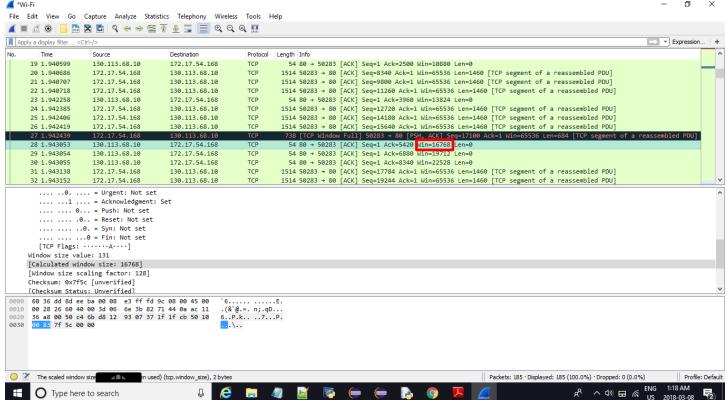


Buffer space: 7936

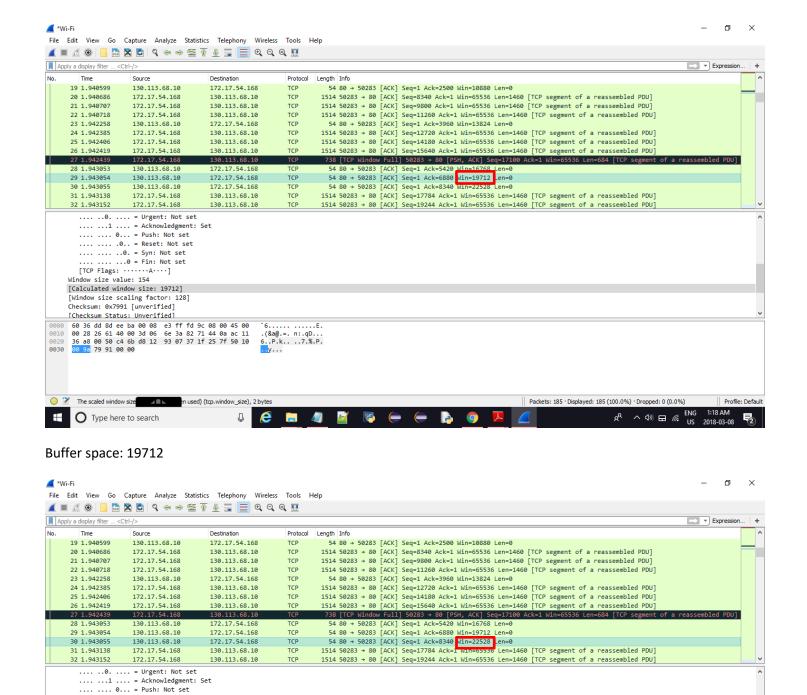


Buffer space: 10880





Buffer Space: 16768



Packets: 185 · Displayed: 185 (100.0%) · Dropped: 0 (0.0%)

x² ^ 40) ⊞ (€

1:18 AM

2018-03-08

US

Buffer space: 22528

O Type here to search

60 36 dd 8d ee ba 00 08 e3 ff fd 9c 08 00 45 00 00 28 26 62 40 00 3d 06 6e 39 82 71 44 0a ac 11 36 a8 00 50 c4 6b d8 12 93 07 37 1f 2b 33 50 10 00 bc 73 c7 00 00

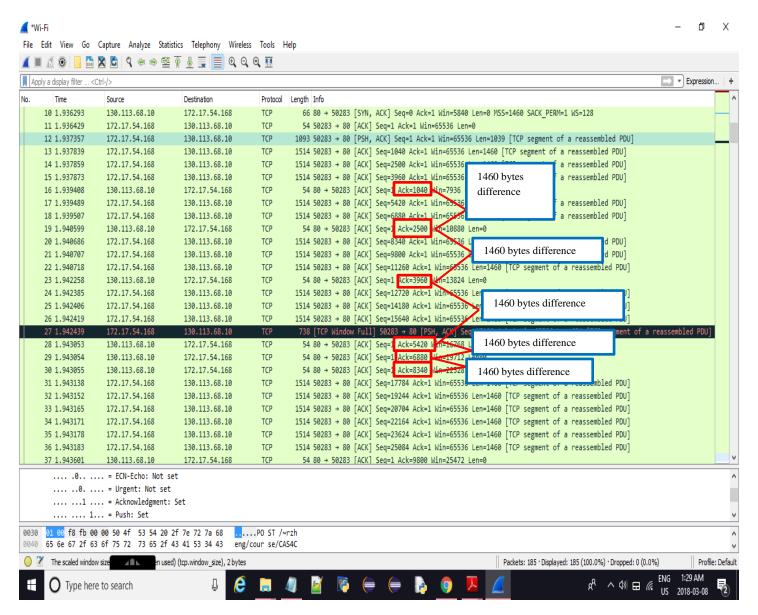
The scaled window size

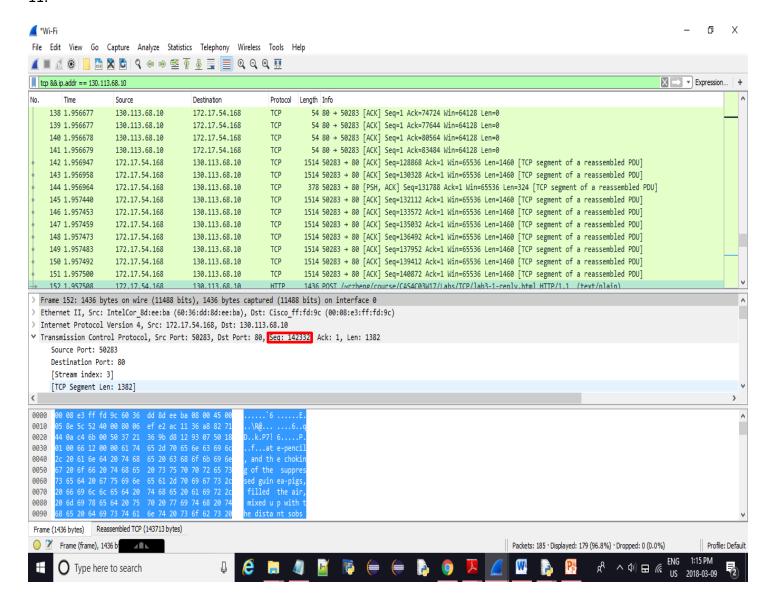
`6.....E. .(&b@.=. n9.qD... 6..P.k....7.+3P. Therefore the minimum buffer space is 7936.

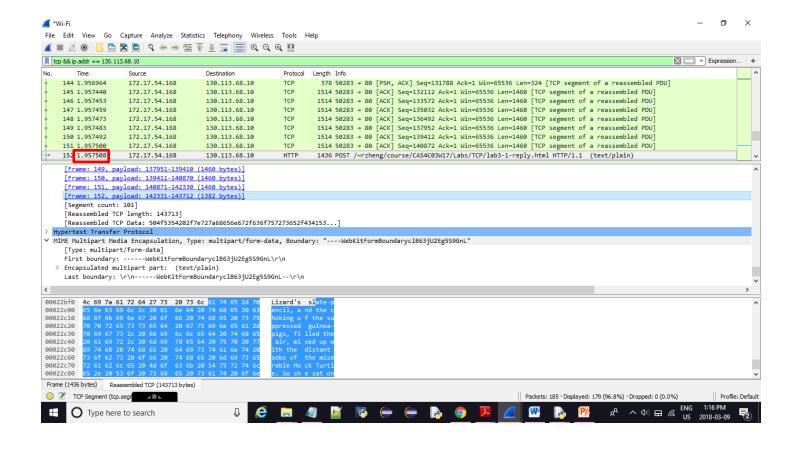
9.

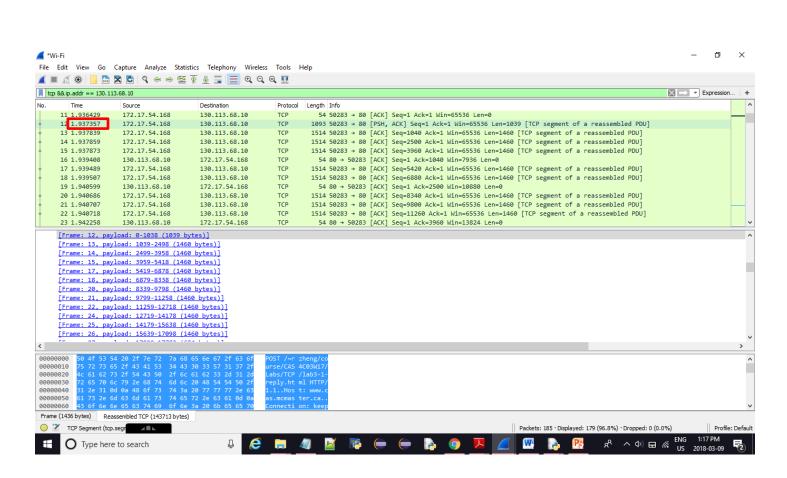
There is no retransimitted segements. I check the ACK value, if there is a retransmitted segment, the ACK value of the received segement should decrease. This can also be checked by the sequence number of the segements being sent, If there is no packets with the same sequence number at different time, then there is no retransmitted segements. In this example, we can not find packets with the same sequence number at different time.











Amount of data transmitted = 142332 bytes

Time incurred = 1.957508 - 1.937357 = 0.020151 s

Throughput = (Amount of data transmitted)/ (time incurred)

= 142332 / 0.020151

= 7063272.294 bytes/sec

4(1)



The TCP slowstart phase starts at 0.006s and ends at 0.02 s. After 0.02s, we do not see the expected linear increase behaviour anymore and TCP session is always in the congestion avoidance state.

The congestion avoidance takes over at 0.013s, 0.02 s, 0.024s, 0.028s. This happens because there are too many cocurrent segements sending at the same time, therefore the rate result in a suddenly drop.