Zachary Waters

Ellen Zegura

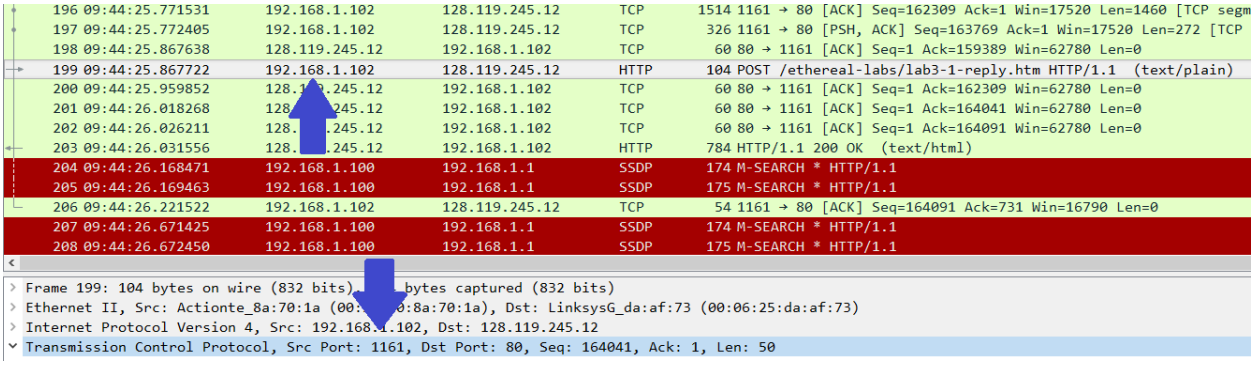
4/22/2019

CS 3251

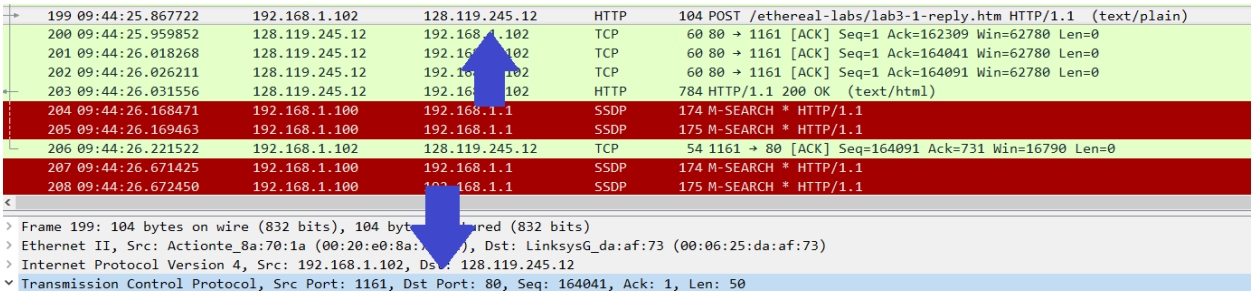
Homework 4:

Wireshark Lab:

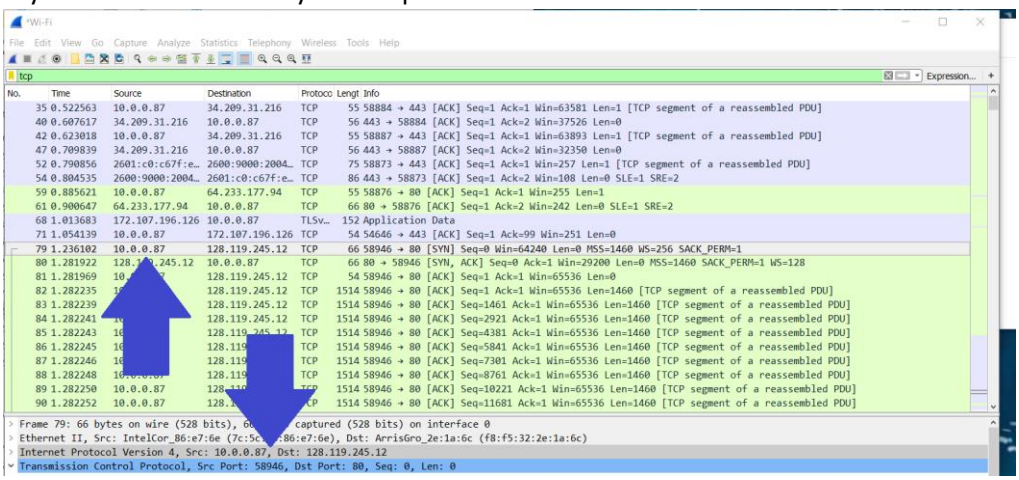
1. The client's IP address is 192.168.1.102 and the TPC port number is 1161.



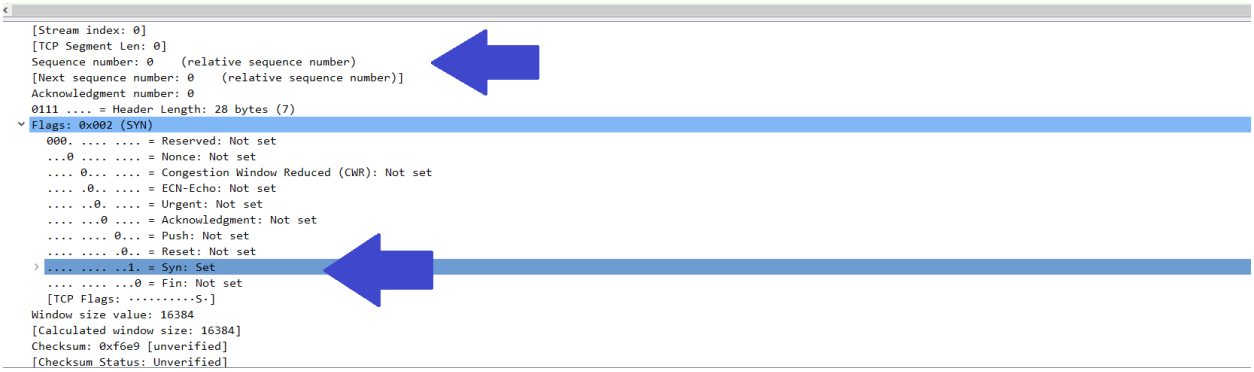
2. the IP address of gaia.cs.umass.edu is 128.119.245.12 and the port number is 80.



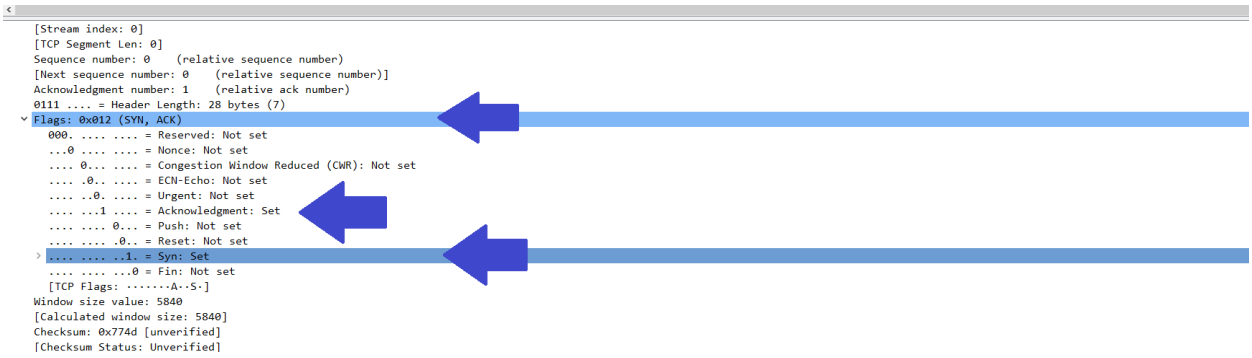
3. My IP is 10.0.0.87 and my source port is 58946



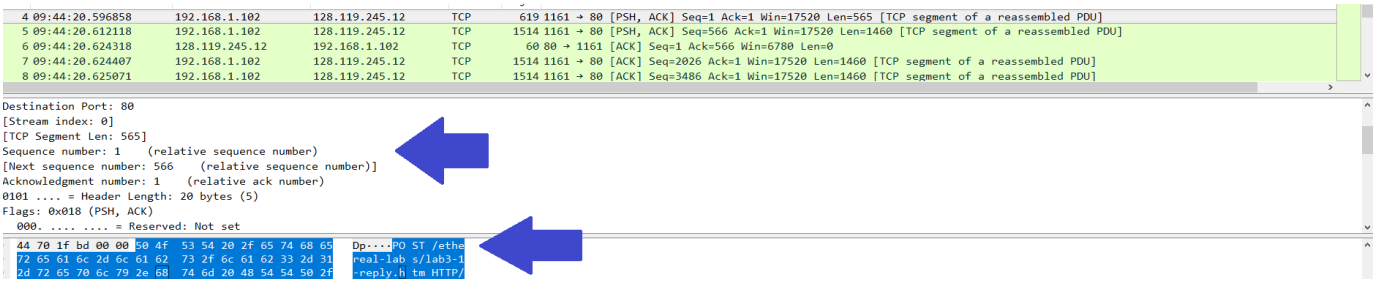
4. The sequence number is 0 and the Syn flag is set to 1 which identifies the segment as a SYN segment.



5. The sequence number of the SYNACK segment is 0, the value of the acknowledgment field is 1, this value is determines by simply adding 1 to the client's original field value. The Syn flag is set to 1 which identifies the segment as a SYN segment.



6. the segment No.4 contains the HTTP POST command, the sequence number of this segment is 1.

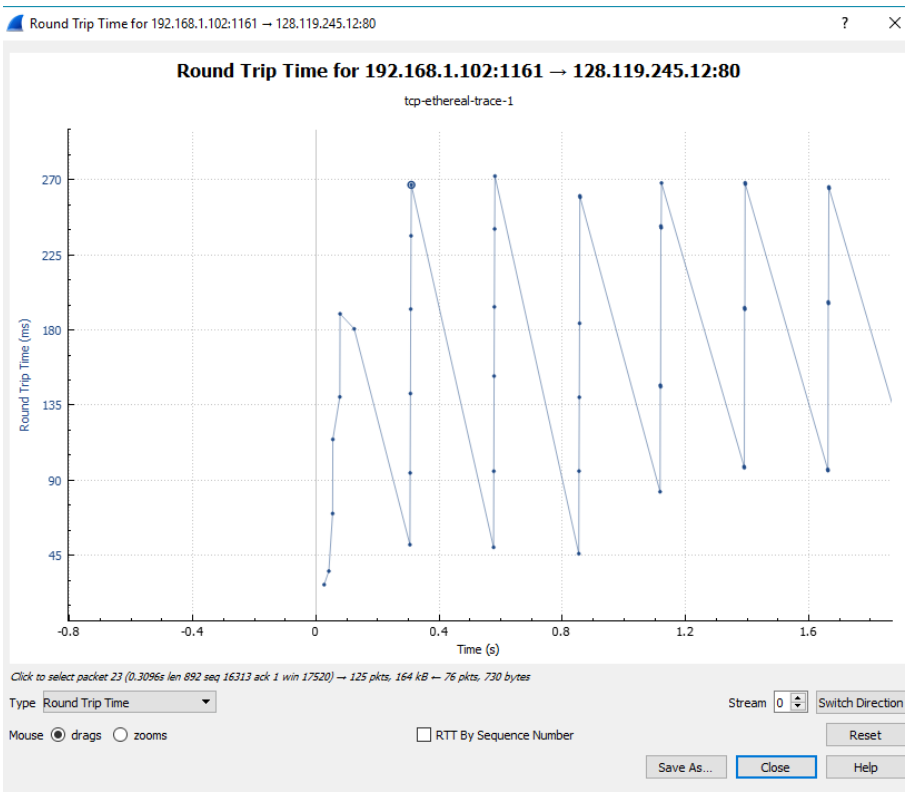


7.

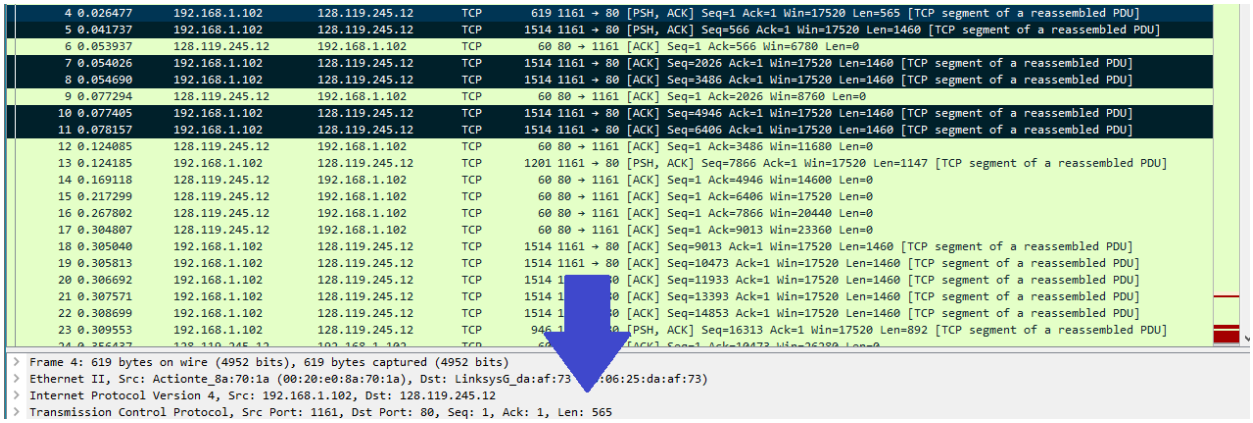
1	0.000000	192.168.1.102	128.119.245.12	TCP	62 1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	128.119.245.12	192.168.1.102	TCP	62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 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=566 Ack=1 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=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=2026 Ack=1 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=3486 Ack=1 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=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=4946 Ack=1 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=6406 Ack=1 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=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 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=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	0.267802	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	0.304807	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18	0.305040	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
19	0.305813	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
20	0.306692	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
21	0.307571	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
22	0.308699	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
23	0.309553	192.168.1.102	128.119.245.12	TCP	946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]

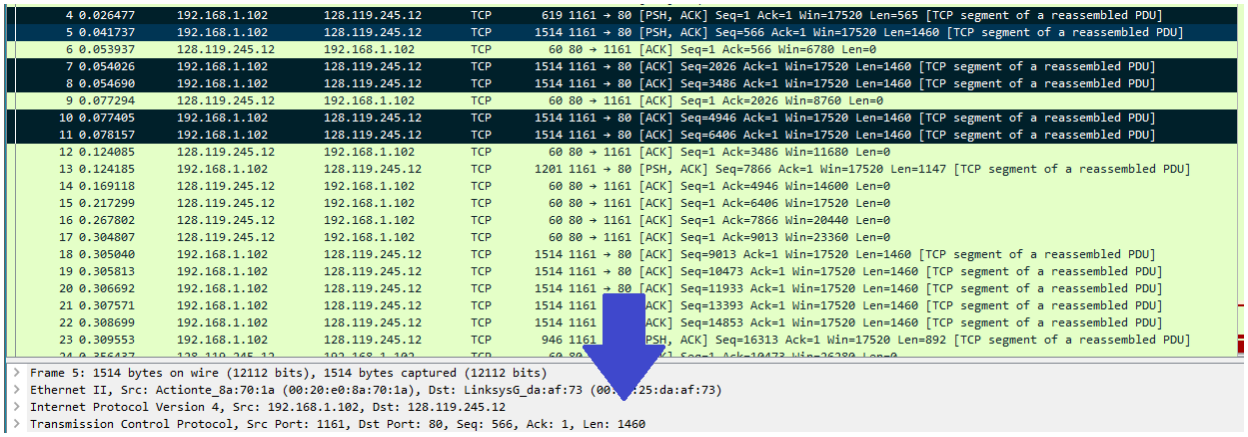
Solution Table:

Segment Number	Wireshark NO.	Sequence Number	Time Sequence	ACK Arrival Time	RTT time	Estimated RTT Value
1	4	1	0.026477	0.053937	0.02746	0.02646
2	5	566	0.041737	0.077294	0.035557	0.0285
3	7	2026	0.054026	0.124085	0.070059	0.0337
4	8	3486	0.05469	0.169118	0.11443	0.0438
5	10	4946	0.077405	0.217299	0.13989	0.0558
6	11	6406	0.078157	0.267802	0.18964	0.0725

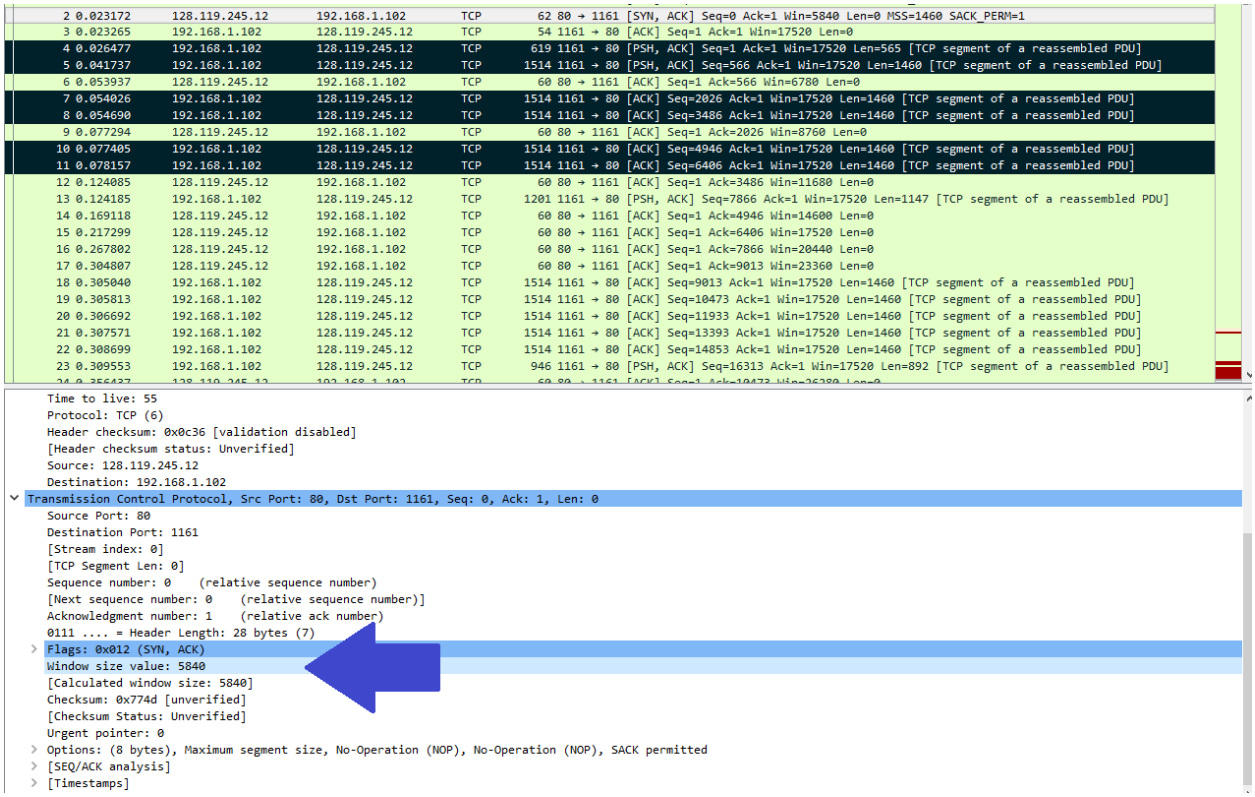


8. The Length of the first TCP segment is 565 bytes, the length of the other 5 segments is 1460 bytes.





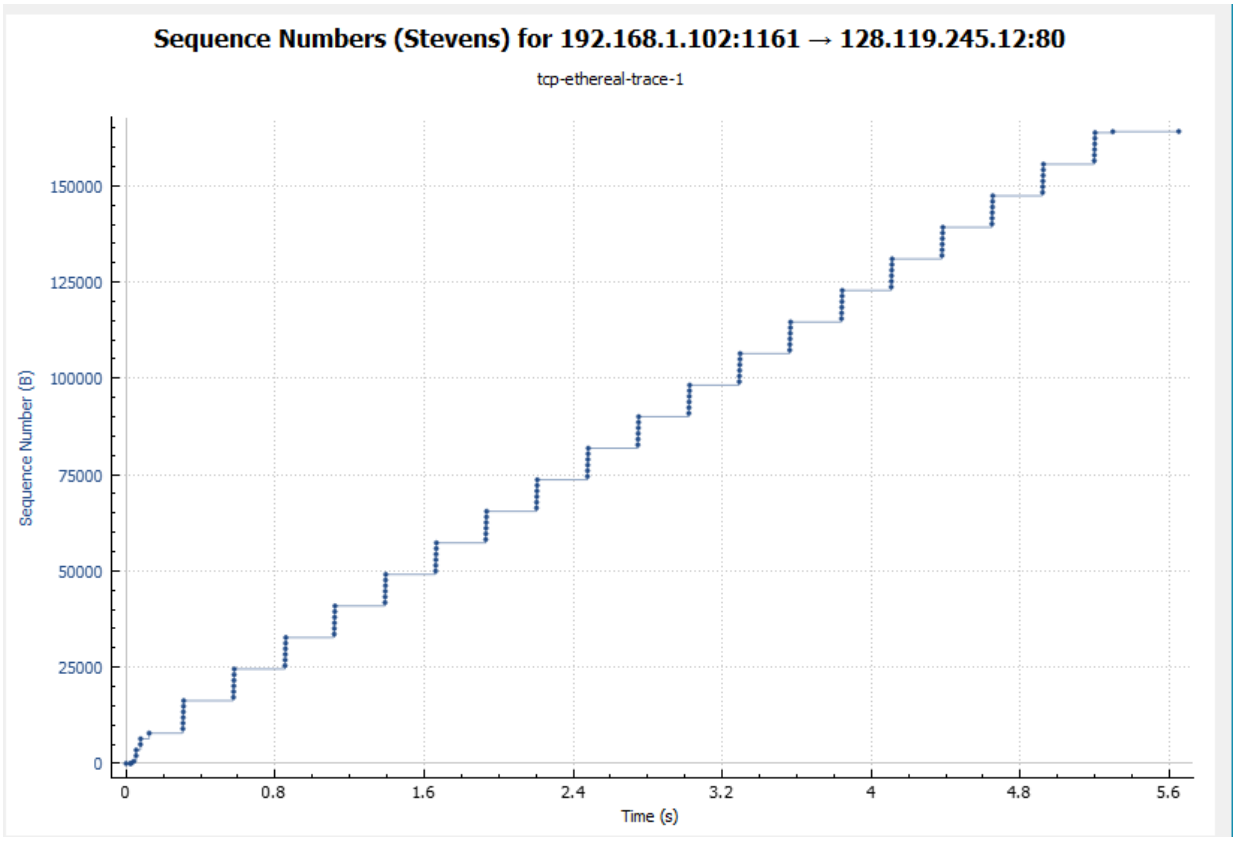
9. The minimum amount of buffer space is 5840 bytes, the sender is never throttled.



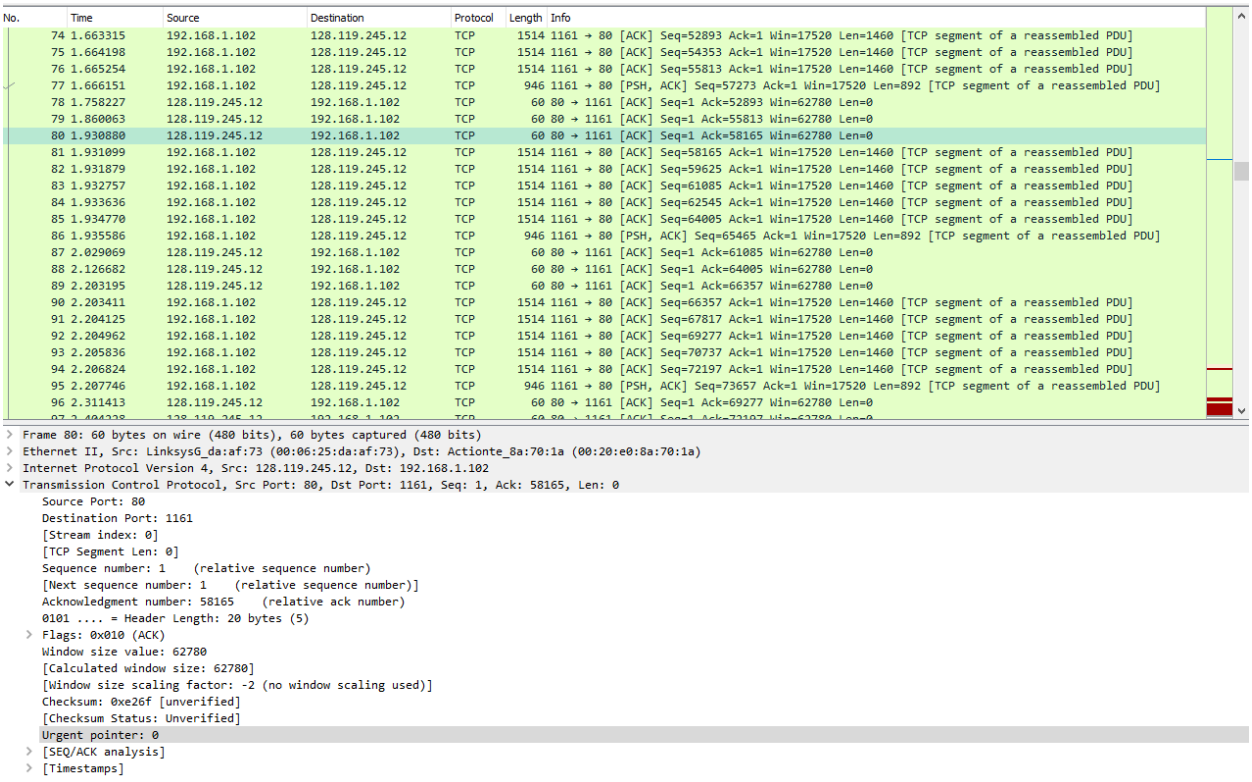
```
 2 0.023172   128.119.245.12   192.168.1.102   TCP    62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1
 3 0.023265   192.168.1.102   128.119.245.12   TCP    54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
 4 0.026477   192.168.1.102   128.119.245.12   TCP   619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 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=566 Ack=1 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=1 Ack=566 Win=6780 Len=0
 7 0.054026   192.168.1.102   128.119.245.12   TCP  1514 1161 → 80 [ACK] Seq=2026 Ack=1 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=3486 Ack=1 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=1 Ack=2026 Win=8760 Len=0
10 0.077405   192.168.1.102   128.119.245.12   TCP  1514 1161 → 80 [ACK] Seq=4946 Ack=1 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=6406 Ack=1 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=1 Ack=3486 Win=11680 Len=0
13 0.124185   192.168.1.102   128.119.245.12   TCP  1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 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=1 Ack=4946 Win=14600 Len=0
15 0.217299   128.119.245.12   192.168.1.102   TCP    60 80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16 0.267802   128.119.245.12   192.168.1.102   TCP    60 80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17 0.304807   128.119.245.12   192.168.1.102   TCP    60 80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18 0.305040   192.168.1.102   128.119.245.12   TCP  1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
19 0.305813   192.168.1.102   128.119.245.12   TCP  1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
20 0.306692   192.168.1.102   128.119.245.12   TCP  1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
21 0.307571   192.168.1.102   128.119.245.12   TCP  1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
22 0.308699   192.168.1.102   128.119.245.12   TCP  1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
23 0.309553   192.168.1.102   128.119.245.12   TCP   946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
```

```
    Time to live: 55
    Protocol: TCP (6)
    Header checksum: 0x0c36 [validation disabled]
    [Header checksum status: Unverified]
    Source: 128.119.245.12
    Destination: 192.168.1.102
▾ Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0
    Source Port: 80
    Destination Port: 1161
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    [Next sequence number: 0    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    0111 .... = Header Length: 28 bytes (7)
  › Flags: 0x012 (SYN, ACK)
    Window size value: 5840
    [Calculated window size: 5840]
    Checksum: 0x774d [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  › Options: (8 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted
  › [SEQ/ACK analysis]
  › [Timestamps]
```

10. There are none, I checked to see if there are any duplicate sequence numbers.



11. The receiver typically acknowledges 1460 bytes of data in an ACK. You can see an example where the receiver acknowledges two segments simultaneously, because packet No. 80 acknowledges 2920 bytes of data.

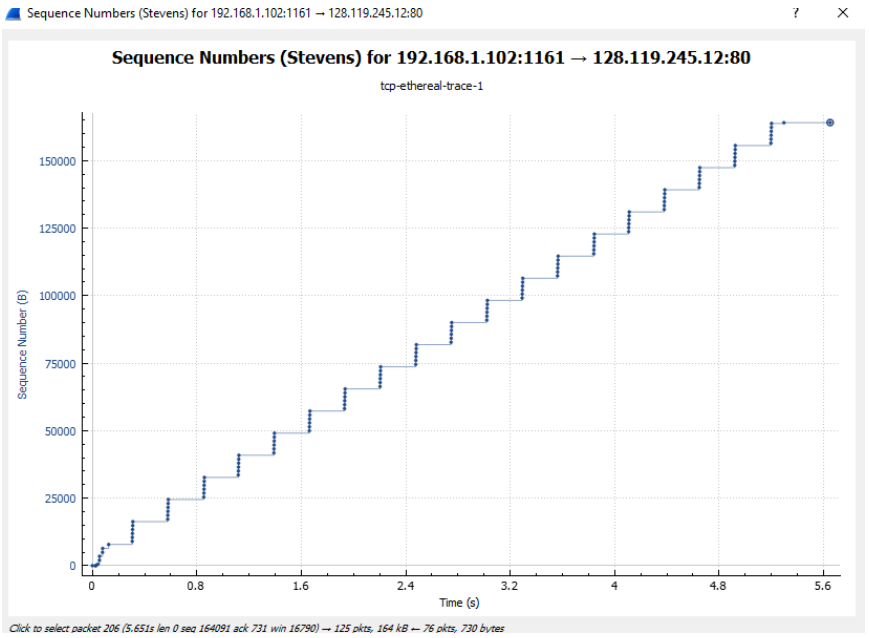


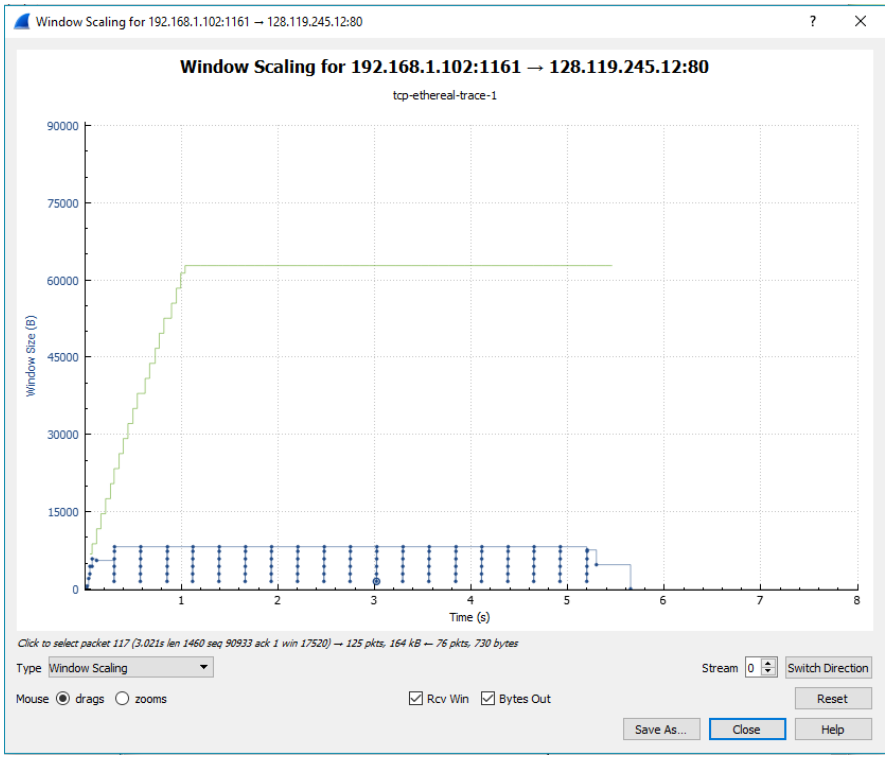
No.	Time	Source	Destination	Protocol	Length	Info
74	1.663315	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=52893 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
75	1.664198	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=54353 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
76	1.665254	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=55813 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
77	1.666151	192.168.1.102	128.119.245.12	TCP	946	1161 → 80 [PSH, ACK] Seq=57273 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
78	1.758227	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=52893 Win=62780 Len=0
79	1.860063	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=55813 Win=62780 Len=0
80	1.930880	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=58165 Win=62780 Len=0
81	1.931099	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=58165 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
82	1.931879	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=59625 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
83	1.932757	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=61085 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
84	1.933636	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=62545 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
85	1.934770	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=64005 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
86	1.935586	192.168.1.102	128.119.245.12	TCP	946	1161 → 80 [PSH, ACK] Seq=65465 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
87	2.029069	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=61085 Win=62780 Len=0
88	2.126682	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=64005 Win=62780 Len=0
89	2.203195	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=66357 Win=62780 Len=0
90	2.203411	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=66357 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
91	2.204125	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=67817 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
92	2.204962	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=69277 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
93	2.205836	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=70737 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
94	2.206824	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=72197 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
95	2.207746	192.168.1.102	128.119.245.12	TCP	946	1161 → 80 [PSH, ACK] Seq=73657 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
96	2.311413	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=69277 Win=62780 Len=0

```
> Frame 80: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
> Ethernet II, Src: LinksysG_da:af:73 (00:06:25:da:af:73), Dst: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
v Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 1, Ack: 58165, Len: 0
    Source Port: 80
    Destination Port: 1161
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 58165    (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window size value: 62780
    [Calculated window size: 62780]
    [Window size scaling factor: -2 (no window scaling used)]
    Checksum: 0xe26f [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  > [SEQ/ACK analysis]
  > [Timestamps]
```

12. The TCP connection starts at No. 4 and ENDS at No. 202. So total time is 5.455830 - 0.026477 = 5.429353. the total bytes sent is No.202's Acknowledgment number - No.4's Sequence number: 164091 - 1 = 164090 bytes. So, the throughput is 30222.7706816 bytes per second.

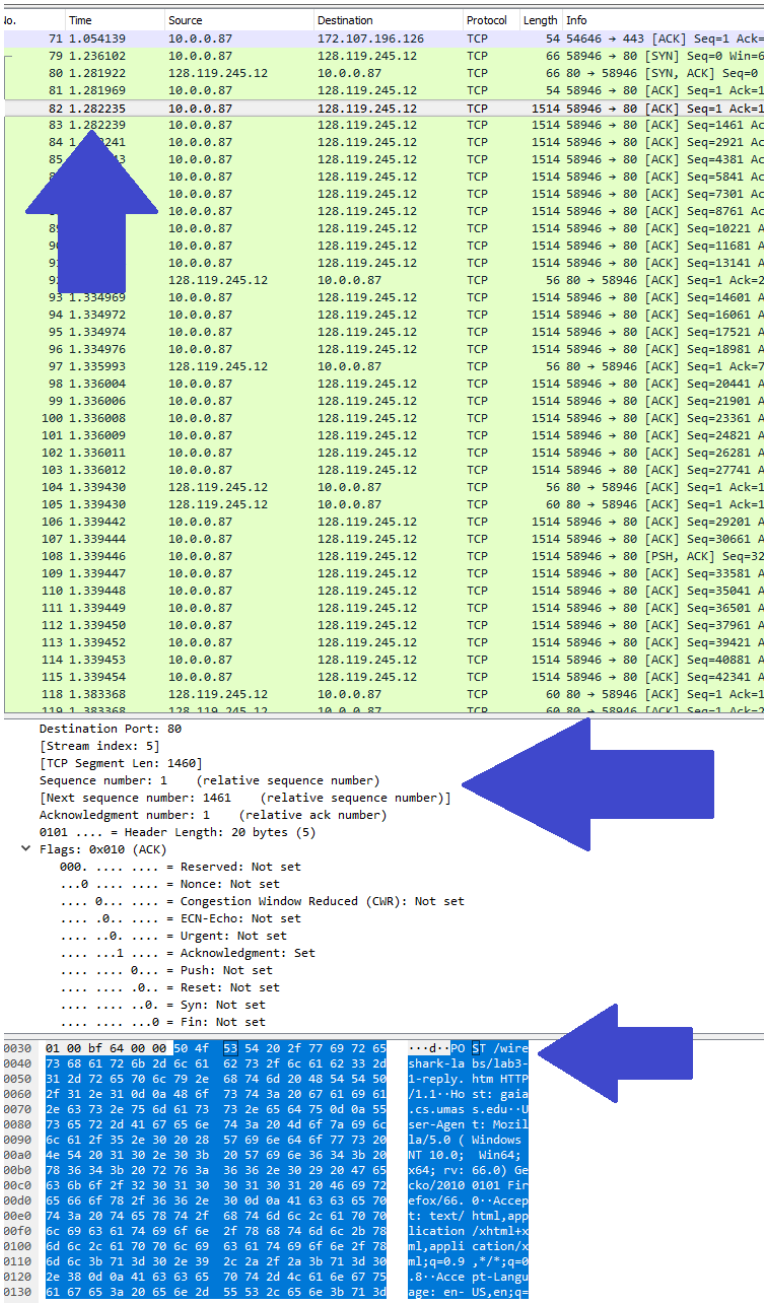
13. TCP's slow start phase begins at time=0 and ends at time=0.1242. the congestion avoidance takes over at time 0.1242 and continues till time 1.03982. and it differs from the idealized behavior due to the "staircase effect" because of the transmission time delays.



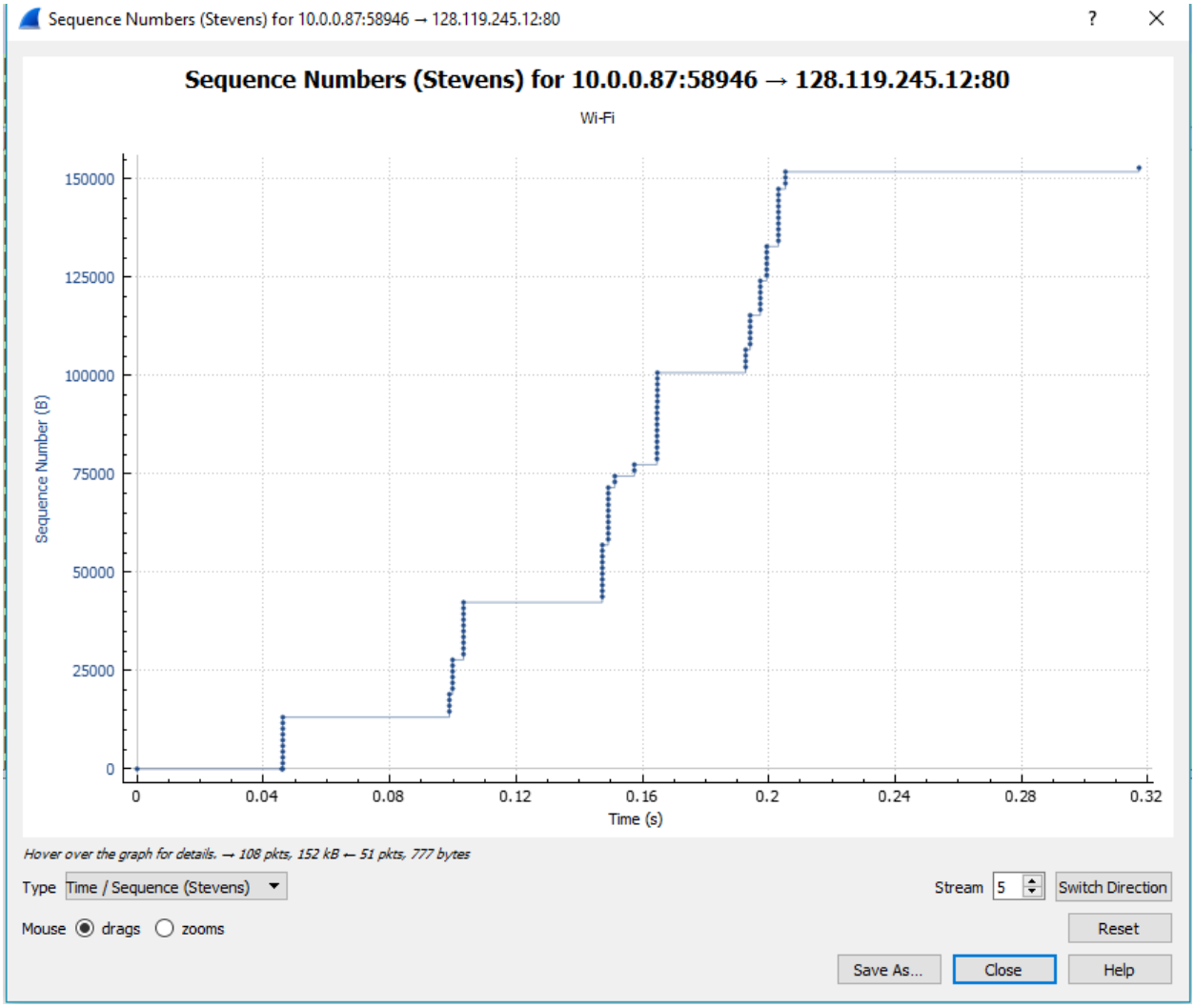


14.

12) the TCP connection starts with packet NO. 82 Time 1.282235 Sequence number 1 and ends with packet No. 246 Time 1.511660 Sequence number 1. This means a total time of 1.511660 - 1.282235 = 0.229425 seconds. Packet N0. 246 had an acknowledgment number of 152873, which when you subtract the final ack by the first sequence number to get the total number of bytes, 152873 - 1 = 152872 bytes. Which mean 666326.686 bytes per second.



13) I cannot identify where my slow start phase begins or ends, using this graph, in addition to the staircase like affect my graph differs from the ideal because it also lacks a slow start phase.



P36:

If the receiver receives a packet with a sequence number that is higher than anticipated, meaning it "missed" a packet, due to packet reordering. In response to detecting this gap, the receiver sends an ack for the initial packet again, and not the out of order one it just received. This continues till it receives the proper in-order packet. TCP is designed to take 3 duplicate acknowledgements to perform a fast retransmission because if waiting to make sure that the in-order packet is actually lost, and not just delayed before retransmitting. This is to prevent the sender from clogging the network with redundant packet.

P40:

a) The TCP slow start is operating between rounds 1-6 and 23-26 because you can see the congestion window size start to ramp up rapidly.

b) The TCP congestion avoidance is operating between rounds 6-16 and 17-22 because you can see the congestion window size increasing linearly.

c) Because the window size was reduced but not all the way dropped to 1, then the loss is due to a triple-duplicate-acknowledgment

d) Because the window size was dropped all the way to 1, then the loss is due to a timeout

e) The value of ssthresh after the first transmission round is 32, since this is the window size reached before switching from slow-start-stop to congestion avoidance.

f) the ssthresh value at round 18 is 21 this is because it is set to half of its size when there is a detected segment loss, and it was recently divided in half from its peak at 42, at round 15, so half of its value is 21.

g) the ssthresh value at round 24 is 14 this is because it is set to half of its size when there is a detected segment loss, and it was recently divided in half from 28, at round 21, so half of its value is 14.

h) the 70^{th} segment was sent during round 7. You can calculate this by summing up the congestion window size of all previous rounds.

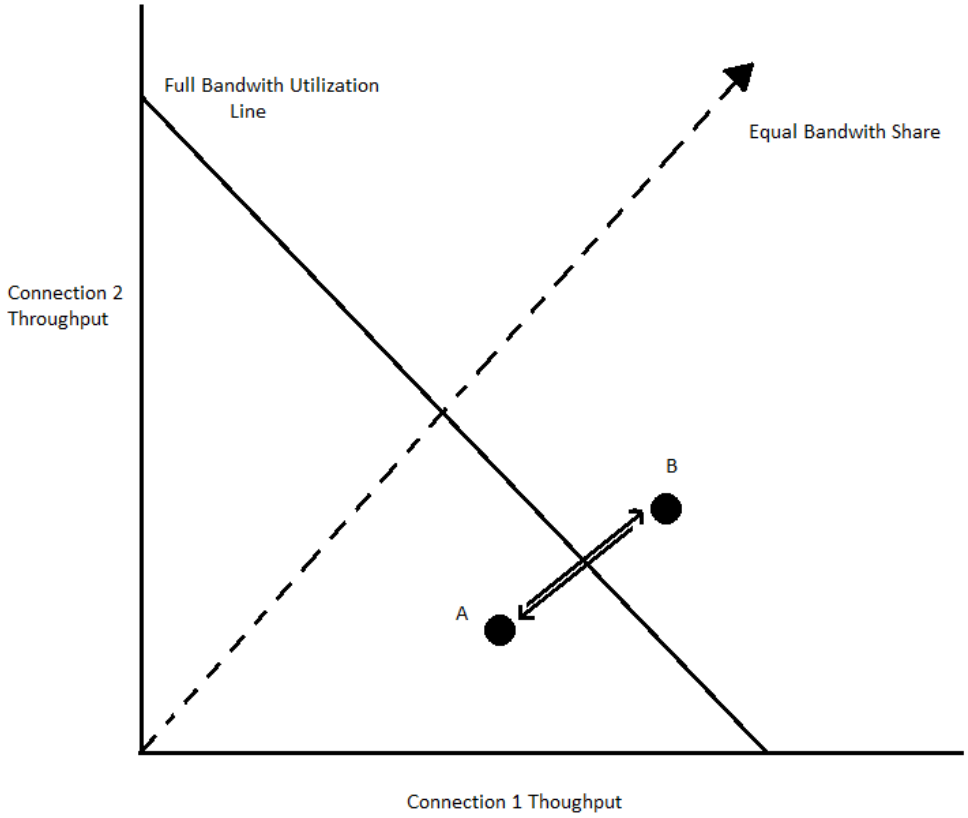
i) the congestion window size and the ssthresh value would halve to 4.

j) the congestion window size would be reset to 1, then regrow to 4 by the 19^{th} round. The ssthresh value would remain the same and be 21.

k) There would be a total of 63 packets, because in round 17, 1 packet is sent. Round 18, 2 packets are sent. Round 19, 4 packets are sent. Round 20, 8 packets are sent. Round 21, 16 packets are sent, and then round 22, 21 packets. So, you have a total of 52.

P41:

It would not converge to an equal share algorithm, simply because if the bandwidth increases increases and decreases with constant size, then if they began at point a, increase by constant size C to point B, on the graph, where they would result in packet lost, and then it would decrease by a constant amount back down to point A. this cycle would continue with the throughput constantly oscillating, thus it would never converge.



P54:

The advantage is that TCP would not have to go through slow start, saving time to ramp up to the throughput value obtained at t1. The disadvantage of using these values is that they may no longer be accurate, because of the delay between t1 and t2 that the sender spends idling. An alternative could be starting a timer for cwnd and ssthresh values, and when this expires, they are reset to their initial values.