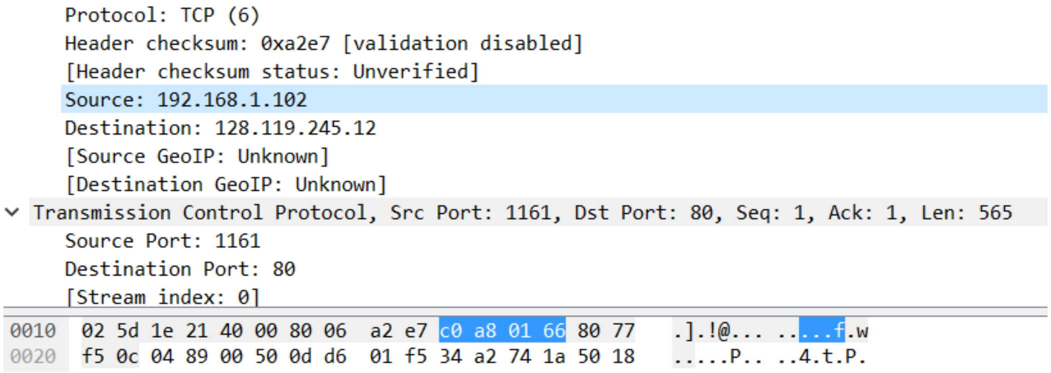
Wireshark Lab #2B: TCP

Suyi Liu, Yuan Jing Vincent Yan

February 22, 2017

Problem 1



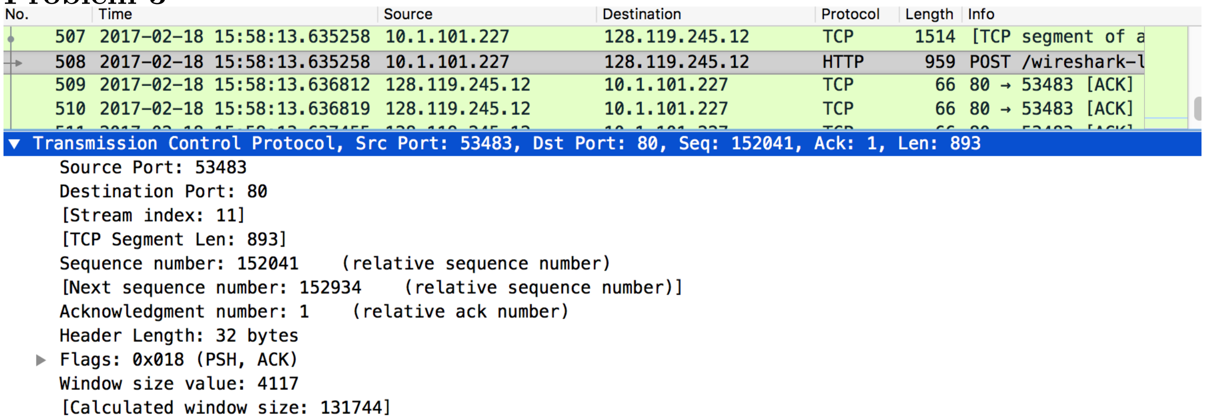
```
      Protocol: TCP (6)
      Header checksum: 0xa2e7 [validation disabled]
      [Header checksum status: Unverified]
      Source: 192.168.1.102
      Destination: 128.119.245.12
      [Source GeoIP: Unknown]
      [Destination GeoIP: Unknown]
  Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 565
      Source Port: 1161
      Destination Port: 80
      [Stream index: 0]
0010  02 5d 1e 21 40 00 80 06  a2 e7 c0 a8 01 66 80 77   .].!@... .....f.w
0020  f5 0c 04 89 00 50 0d d6  01 f5 34 a2 74 1a 50 18   .....P.. ..4.t.P.
```

The IP address and TCP port number used by the client computer that is transferring the file to gaia.cs.umass.edu is 192.168.1.102 and 1161 respectively.

Problem 2

(Referring to the same screenshot as problem 1)
The IP address of gaia.cs.umass.edu is 128.119.245.12
On port 80 it is sending and receiving TCP segments for this connection.

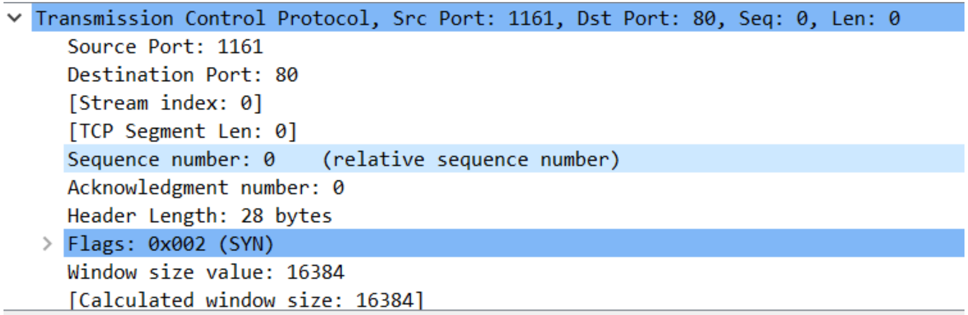
Problem 3



```
No.   Time                        Source          Destination     Protocol  Length  Info
  507 2017-02-18 15:58:13.635258 10.1.101.227    128.119.245.12  TCP         1514  [TCP segment of a
  508 2017-02-18 15:58:13.635258 10.1.101.227    128.119.245.12  HTTP         959  POST /wireshark-l
  509 2017-02-18 15:58:13.636812 128.119.245.12  10.1.101.227    TCP           66  80 → 53483 [ACK]
  510 2017-02-18 15:58:13.636819 128.119.245.12  10.1.101.227    TCP           66  80 → 53483 [ACK]
  511 2017-02-18 15:58:13.637455 128.119.245.12  10.1.101.227    TCP           66  80 → 53483 [ACK]
▼ Transmission Control Protocol, Src Port: 53483, Dst Port: 80, Seq: 152041, Ack: 1, Len: 893
     Source Port: 53483
     Destination Port: 80
     [Stream index: 11]
     [TCP Segment Len: 893]
     Sequence number: 152041    (relative sequence number)
     [Next sequence number: 152934    (relative sequence number)]
     Acknowledgment number: 1    (relative ack number)
     Header Length: 32 bytes
   ▶ Flags: 0x018 (PSH, ACK)
     Window size value: 4117
     [Calculated window size: 131744]
```

The public IP address used by my client computer is: 96.91.196.137
The private IP address(shown on Wireshark) used by my client computer is: 10.1.101.227
The port number used by my client computer is: 53483

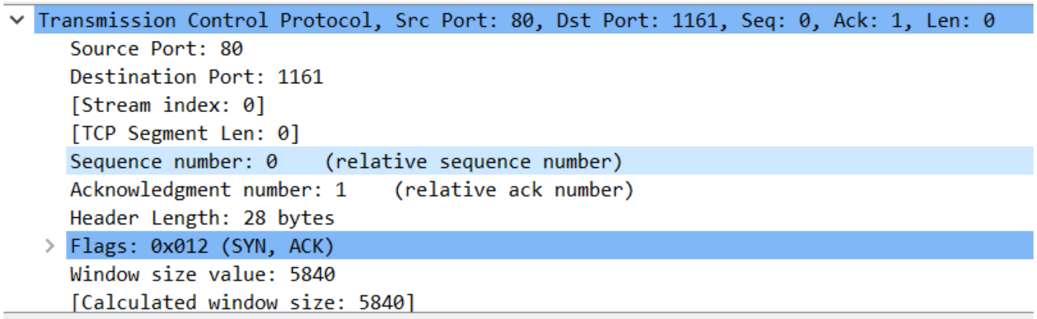
Problem 4



```
⌄ Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
     Source Port: 1161
     Destination Port: 80
     [Stream index: 0]
     [TCP Segment Len: 0]
     Sequence number: 0    (relative sequence number)
     Acknowledgment number: 0
     Header Length: 28 bytes
   > Flags: 0x002 (SYN)
     Window size value: 16384
     [Calculated window size: 16384]
```

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 0.
The flags are set to be 0x002(The SYN flag is set to 1) identifying the segment as a SYN message.

Problem 5



```
v 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)
    Acknowledgment number: 1    (relative ack number)
    Header Length: 28 bytes
  > Flags: 0x012 (SYN, ACK)
    Window size value: 5840
    [Calculated window size: 5840]
```

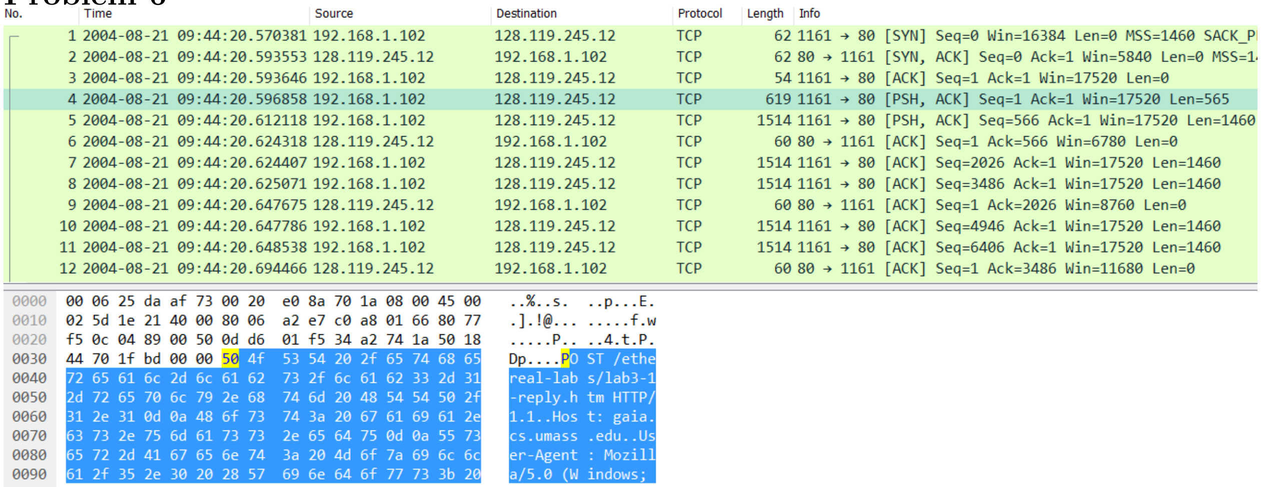
a. The sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN is 0.

b. The value of the Acknowledgement field in the SYNACK segment is 1.

c. gaia.cs.umass.edu determines that value by setting it to client_isn+1, where client_isn is the sequence number of the TCP SYN segment sent by the client earlier.

d. The flags are set as 0x012(The SYN flag and Acknowledgement flag are both set to 1) in the segment identifying the segment as a SYNACK segment.

Problem 6

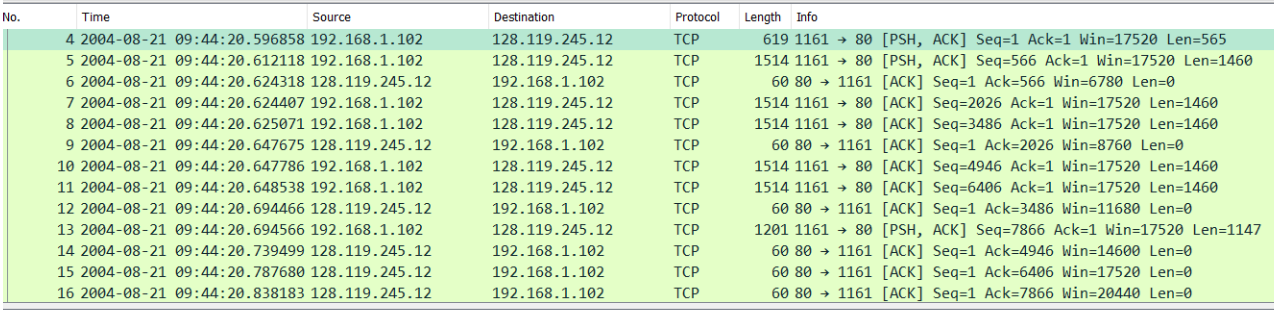


```
No.     Time                        Source          Destination     Protocol  Length  Info
      1 2004-08-21 09:44:20.570381 192.168.1.102   128.119.245.12   TCP         62 1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_P
      2 2004-08-21 09:44:20.593553 128.119.245.12  192.168.1.102    TCP         62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1
      3 2004-08-21 09:44:20.593646 192.168.1.102   128.119.245.12   TCP         54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
      4 2004-08-21 09:44:20.596858 192.168.1.102   128.119.245.12   TCP        619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565
      5 2004-08-21 09:44:20.612118 192.168.1.102   128.119.245.12   TCP       1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460
      6 2004-08-21 09:44:20.624318 128.119.245.12  192.168.1.102    TCP         60 80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
      7 2004-08-21 09:44:20.624407 192.168.1.102   128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460
      8 2004-08-21 09:44:20.625071 192.168.1.102   128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460
      9 2004-08-21 09:44:20.647675 128.119.245.12  192.168.1.102    TCP         60 80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
     10 2004-08-21 09:44:20.647786 192.168.1.102   128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460
     11 2004-08-21 09:44:20.648538 192.168.1.102   128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460
     12 2004-08-21 09:44:20.694466 128.119.245.12  192.168.1.102    TCP         60 80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
```

```
0000  00 06 25 da af 73 00 20  e0 8a 70 1a 08 00 45 00   ..%..s.  ..p...E.
0010  02 5d 1e 21 40 00 80 06  a2 e7 c0 a8 01 66 80 77   .].!@... .....f.w
0020  f5 0c 04 89 00 50 0d d6  01 f5 34 a2 74 1a 50 18   .....P.. ..4.t.P.
0030  44 70 1f bd 00 00 50 4f  53 54 20 2f 65 74 68 65   Dp....PO ST /ethe
0040  72 65 61 6c 2d 6c 61 62  73 2f 6c 61 62 33 2d 31   real-lab s/lab3-1
0050  2d 72 65 70 6c 79 2e 68  74 6d 20 48 54 54 50 2f   -reply.h tm HTTP/
0060  31 2e 31 0d 0a 48 6f 73  74 3a 20 67 61 69 61 2e   1.1..Hos t: gaia.
0070  63 73 2e 75 6d 61 73 73  2e 65 64 75 0d 0a 55 73   cs.umass .edu..Us
0080  65 72 2d 41 67 65 6e 74  3a 20 4d 6f 7a 69 6c 6c   er-Agent : Mozill
0090  61 2f 35 2e 30 20 28 57  69 6e 64 6f 77 73 3b 20   a/5.0 (W indows;
```

The sequence number of the TCP segment containing the HTTP POST command is 1.

Problem 7



No.	Time	Source	Destination	Protocol	Length	Info
4	2004-08-21 09:44:20.596858	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565
5	2004-08-21 09:44:20.612118	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460
6	2004-08-21 09:44:20.624318	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	2004-08-21 09:44:20.624407	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460
8	2004-08-21 09:44:20.625071	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460
9	2004-08-21 09:44:20.647675	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	2004-08-21 09:44:20.647786	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460
11	2004-08-21 09:44:20.648538	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460
12	2004-08-21 09:44:20.694466	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	2004-08-21 09:44:20.694566	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147
14	2004-08-21 09:44:20.739499	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	2004-08-21 09:44:20.787680	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	2004-08-21 09:44:20.838183	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0

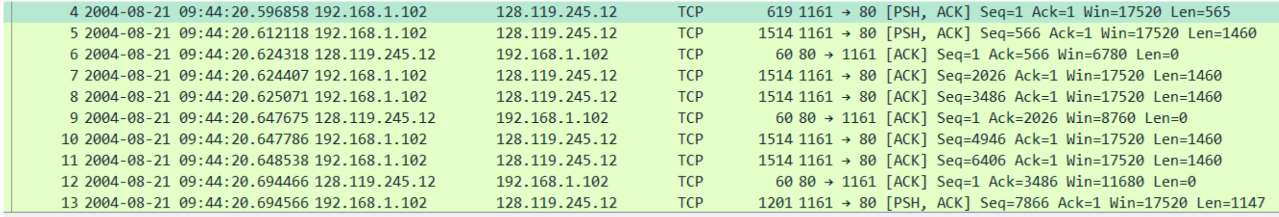
a. The sequence numbers of the first six segments in the TCP connection(including the segment containing the HTTP POST) are: 1,566,2026,3486,4946,6406.

b. Each segment was sent at 2004-08-21 09:44:20.596858, 2004-08-21 09:44:20.612118, 2004-08-21 09:44:20.624407, 2004-08-21 09:44:20.625071, 2004-08-21 09:44:20.647786, 2004-08-21 09:44:20.648538 respectively.

c. The ACK for each segment was received at 2004-08-21 09:44:20.624318, 2004-08-21 09:44:20.647675, 2004-08-21 09:44:20.694466, 2004-08-21 09:44:20.739499, 2004-08-21 09:44:20.787680, 2004-08-21 09:44:20.838183 respectively.

d. The RTT value for each of the six segments are: 0.02746, 0.035557, 0.070059, 0.11443, 0.13989, 0.18964. (in Second)
EstimatedRTT = 0.875 *EstimatedRTT + 0.125*SampleRTT
The EstimatedRTT value after the receipt of each ACK are: 0.02746, 0.0285, 0.0337, 0.0438, 0.0558, 0.0725. (in Second)

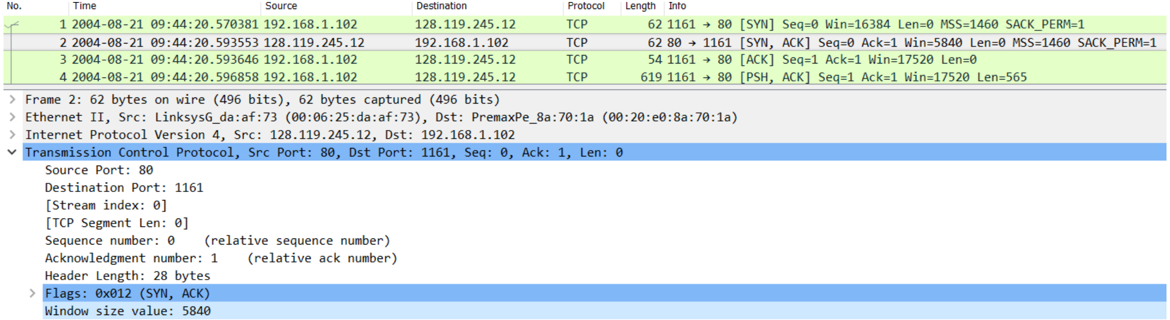
Problem 8



4	2004-08-21 09:44:20.596858	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565
5	2004-08-21 09:44:20.612118	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460
6	2004-08-21 09:44:20.624318	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	2004-08-21 09:44:20.624407	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460
8	2004-08-21 09:44:20.625071	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460
9	2004-08-21 09:44:20.647675	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	2004-08-21 09:44:20.647786	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460
11	2004-08-21 09:44:20.648538	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460
12	2004-08-21 09:44:20.694466	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	2004-08-21 09:44:20.694566	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147

The length of each of the first six TCP segments are: 565 bytes, 1460 bytes, 1460 bytes, 1460 bytes, 1460 bytes, 1460 bytes, respectively. (Look at the "Len=xxx" field in the list of captured packets)

Problem 9



```
No.   Time                        Source          Destination     Protocol Length Info
    1 2004-08-21 09:44:20.570381 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 2004-08-21 09:44:20.593553 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 2004-08-21 09:44:20.593646 192.168.1.102   128.119.245.12  TCP        54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
    4 2004-08-21 09:44:20.596858 192.168.1.102   128.119.245.12  TCP       619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565
```

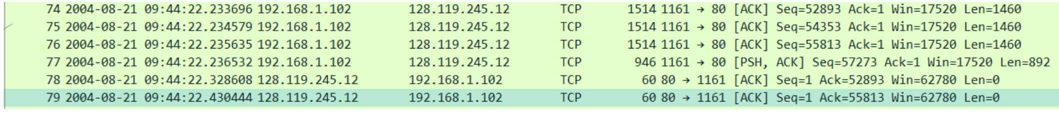
```
> Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
> Ethernet II, Src: LinksysG_da:af:73 (00:06:25:da:af:73), Dst: PremaxPe_8a:70:1a (00:20:e0:8a:70:1a)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
v Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0
    Source Port: 80
    Destination Port: 1161
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Acknowledgment number: 1    (relative ack number)
    Header Length: 28 bytes
  > Flags: 0x012 (SYN, ACK)
    Window size value: 5840
```

The minimum amount of available buffer space advertised at the received for the entire trace is 5840 bytes.
The lack of receiver buffer space doesn't ever throttle the sender.

Problem 10

There aren't any retransmitted segments in the trace file. We found this by checking the sequence numbers from the source to the destination in the whole file. Specifically, the sequence numbers are increasing monotonically without repetitions.

Problem 11



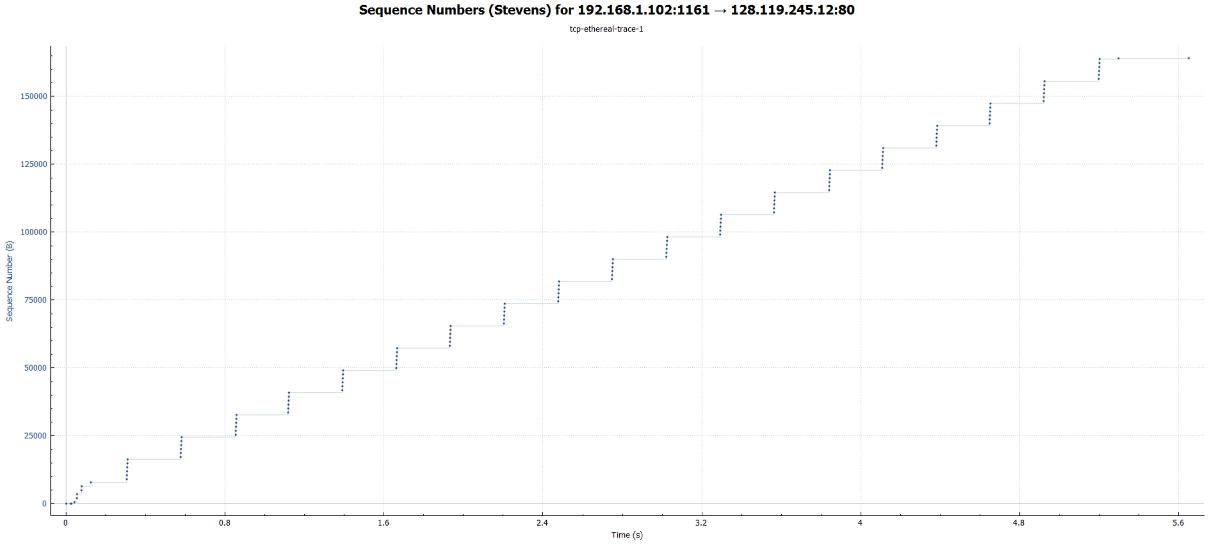
```
   74 2004-08-21 09:44:22.233696 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=52893 Ack=1 Win=17520 Len=1460
   75 2004-08-21 09:44:22.234579 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=54353 Ack=1 Win=17520 Len=1460
   76 2004-08-21 09:44:22.235635 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=55813 Ack=1 Win=17520 Len=1460
   77 2004-08-21 09:44:22.236532 192.168.1.102   128.119.245.12  TCP       946 1161 → 80 [PSH, ACK] Seq=57273 Ack=1 Win=17520 Len=892
   78 2004-08-21 09:44:22.328608 128.119.245.12  192.168.1.102   TCP        60 80 → 1161 [ACK] Seq=1 Ack=52893 Win=62780 Len=0
   79 2004-08-21 09:44:22.430444 128.119.245.12  192.168.1.102   TCP        60 80 → 1161 [ACK] Seq=1 Ack=55813 Win=62780 Len=0
```

In most cases, the receiver typically acknowledge 1460 bytes in an ACK (sometimes less, such as 566 or 1147 bytes).
Case where the receiver is ACKing every other received segment: segment 78 ACKs sequence number 52893, and segment 79 ACKs sequence number 55813, but skipped ACKing 54353 in response to segment 74.

Problem 12

The throughput for the TCP connection is computed by dividing total amount of data transferred(the sequence number of the last ACK(sequence no.202) - the sequence number of the first TCP segment(sequence no.4)) by total amount of transmission time.
Total amount of data transferred are 164091 - 1 = 164090 bytes
Total transmission time taken: 5.455830 - 0.026477 = 5.4294 seconds
Throughput for the TCP connection is: 164090/5.4294 = 30.222 KByte/second

Problem 13



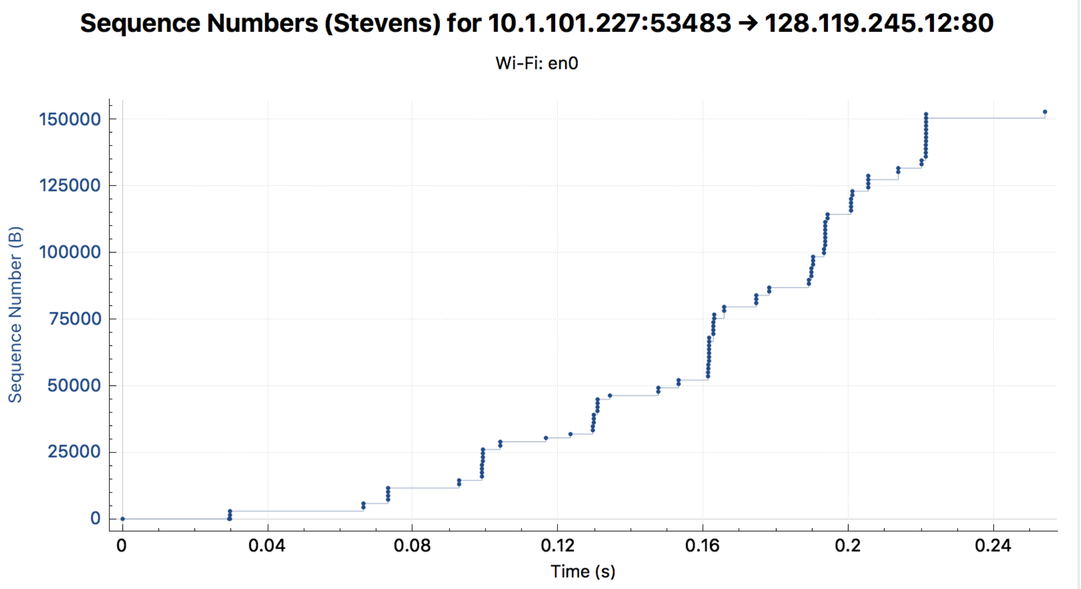
TCP's slow start phase begins as the segment containing HTTP POST was sent. We can identify where TCP slow start phases and congestion avoidance phases are by examining the value of the congestion window size of this TCP sender. Although we cannot obtain the congestion window size directly, the amount of data stacked at a time on the Time-Sequence-Graph (Stevens) graph provides an lower bound of the window size.

From this plot, we can see that for the first 0.4 seconds TCP is in slow start, since the window is growing approximately exponentially. Then, from 0.4 seconds until the end (5.6 seconds), we can see that TCP is in congestion avoidance where the window is growing linearly very slowly, almost constantly.

Comment on ways in which the measured data differs from the idealized behavior of TCP that weve studied in the text:

The idealized behavior of TCP follows a saw-tooth shape. However, our measure data is rather constant throughout the entire transmission expect for the exponentially growth in the beginning. This is likely due to that we are not trasnmitting that many data, so our entire transmission behaves like just the spiking on the first tooth in the saw-tooth shape. If we were to continue transmitting a lot of data, the shape should come to resemble that of the idealized behavior.

Problem 14



Same as in Problem 13, we can use the amount of data stacked at a time on the Time-Sequence-Graph (Stevens) graph as an lower bound to estimate the window size. From this plot, we can see that the transmission follows roughly a saw-tooth shape. We can see the "tooth" spikes at around 1.0sec, 1.3sec, 1.6sec, 1.9sec and 2.2sec.

Comment on ways in which the measured data differs from the idealized behavior of TCP that weve studied in the text:

Our measure data is quite similar to the idealized saw-tooth behavior except that it is not as smooth, since network has fluctuations, albeit little ones, all the time.