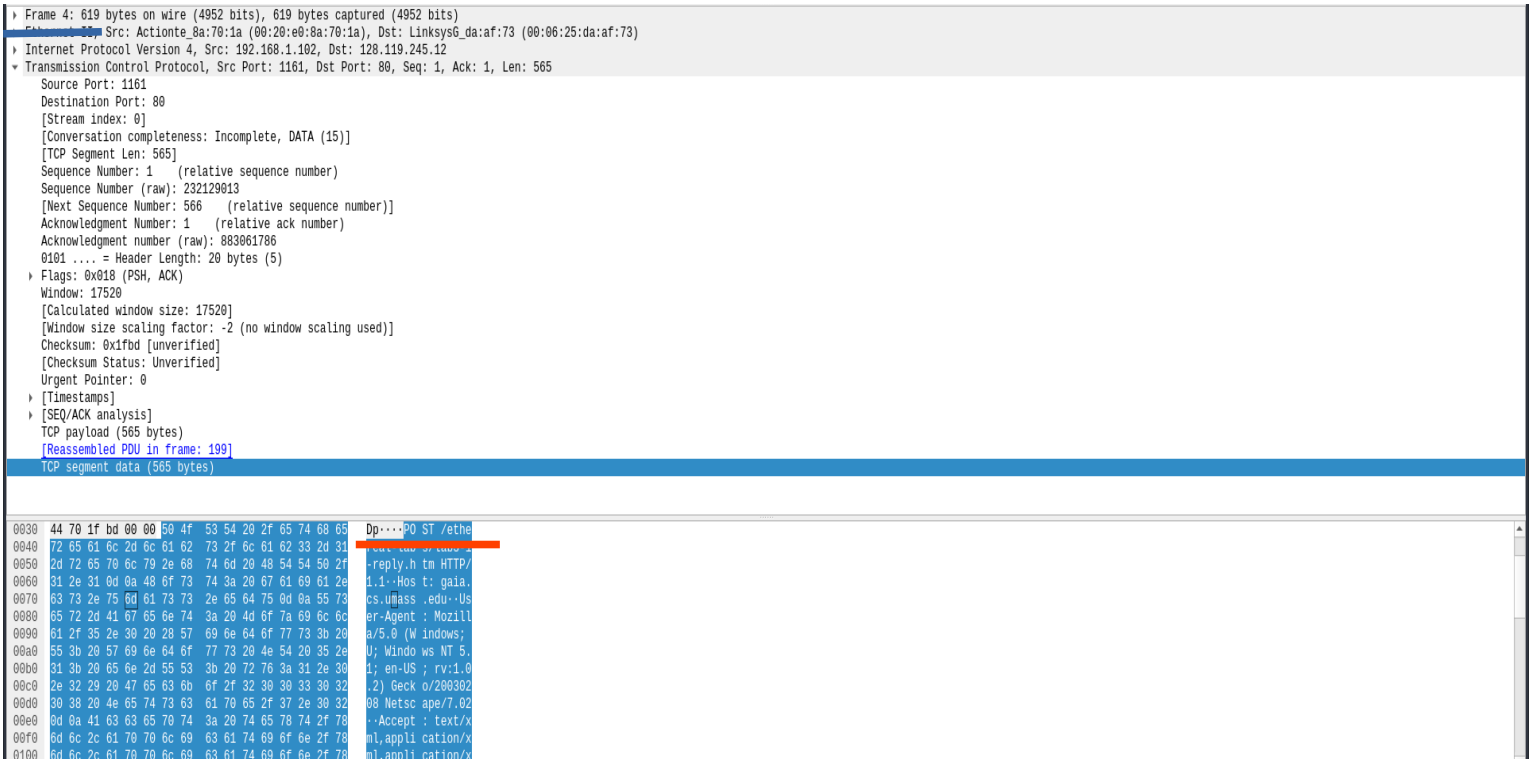
LAB 3

TDTS04

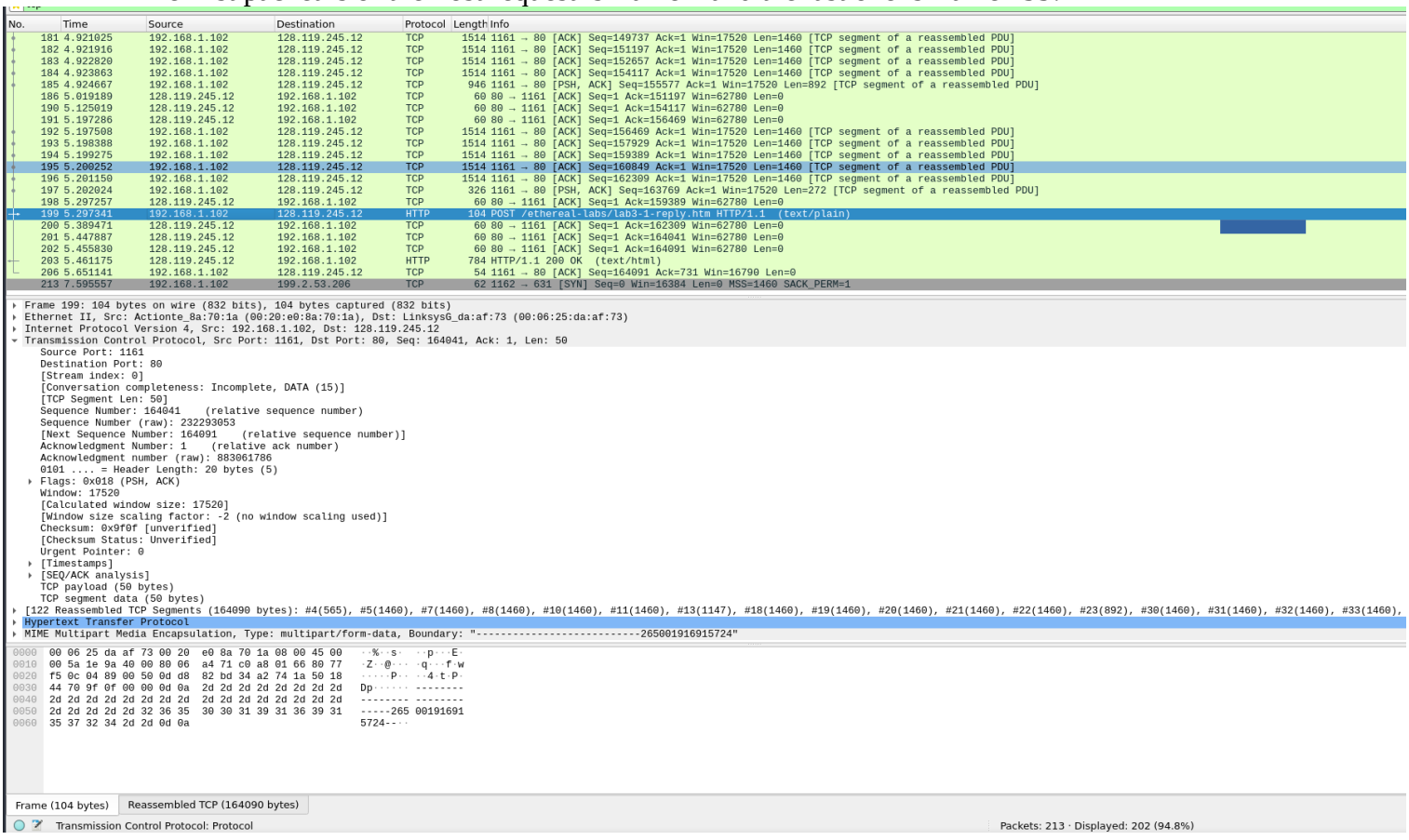
hugho678 & viles222

Answer the following questions (short answers):

1. What are the first and last packets for the POST request? Hint: Look for where the POST request is first initiated in the TCP stream; it often starts earlier in the trace than where 'POST' is explicitly labeled.



The first packet is of the Post request is frame 4 and the last one is frame 199.



2. What is the IP address and the TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu?

/home/viles222/Documents/wireshark/tcp-ethereal-trace-1 213 total packets, 213 shown

```
No.     Time           Source                Destination           Protocol Length Info
     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
Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
    Destination: LinksysG_da:af:73 (00:06:25:da:af:73)
        Address: LinksysG_da:af:73 (00:06:25:da:af:73)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Source: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
        Address: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 232129012
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 0
    Acknowledgment number (raw): 0
    0111 .... = Header Length: 28 bytes (7)
    Flags: 0x002 (SYN)
    Window: 16384
    [Calculated window size: 16384]
    Checksum: 0xf6e9 [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
    Options: (8 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted
    [Timestamps]
```

The source ip address is 192.168.1.102 and the sending and reciving port number is 1161 as seen in blue.

3. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?

The ip address of gaia.cs.umass.edu is 128.119.245.12 and the port being used is 80 as seen in red.

4. What is 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? What is it in the segment that identifies the segment as a SYN segment?

```
No.     Time           Source                Destination           Protocol Length Info
      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
Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 232129012
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 0
    Acknowledgment number (raw): 0
    0111 .... = Header Length: 28 bytes (7)
    Flags: 0x002 (SYN)
        000. .... .... = Reserved: Not set
        ...0 .... .... = Nonce: Not set
        .... 0... .... = Congestion Window Reduced (CWR): Not set
        .... .0.. .... = ECN-Echo: Not set
        .... ..0. .... = Urgent: Not set
        .... ...0 .... = Acknowledgment: Not set
        .... .... 0... = Push: Not set
        .... .... .0.. = Reset: Not set
        .... .... ..1. = Syn: Set
        .... .... ...0 = Fin: Not set
        [TCP Flags: ··········S·]
    Window: 16384
    [Calculated window size: 16384]
    Checksum: 0xf6e9 [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
    Options: (8 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted
    [Timestamps]
```

The sequence number for the TCP SYN segment is 232129012 and in the segment there is a syn flag set to identify it as a SYN segment as seen in blue.

5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the ACKnowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?

/home/viles222/Documents/wireshark/tcp-ethereal-trace-1 213 total packets, 202 shown

```
No.     Time           Source                Destination           Protocol Length Info
      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
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: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 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]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 883061785
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 232129013
    0111 .... = Header Length: 28 bytes (7)
    Flags: 0x012 (SYN, ACK)
        000. .... .... = Reserved: Not set
        ...0 .... .... = Nonce: Not set
        .... 0... .... = Congestion Window Reduced (CWR): Not set
        .... .0.. .... = ECN-Echo: Not set
        .... ..0. .... = Urgent: Not set
        .... ...1 .... = Acknowledgment: Set
        .... .... 0... = Push: Not set
        .... .... .0.. = Reset: Not set
        .... .... ..1. = Syn: Set
        .... .... ...0 = Fin: Not set
        [TCP Flags: ·····A··S·]
    Window: 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
    [Timestamps]
    [SEQ/ACK analysis]
```

The sequence number sent by the server is 883061785. In the SYNACK segment the ACK value is 232129013, which is the TCP syn value incremented once as to identify it as a synack segment

6. What is the sequence number of the TCP segment containing the HTTP POST command?

```
No.     Time           Source                Destination           Protocol Length Info
      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]
Frame 4: 619 bytes on wire (4952 bits), 619 bytes captured (4952 bits)
Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 565
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 565]
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 232129013
    [Next Sequence Number: 566    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 883061786
    0101 .... = Header Length: 20 bytes (5)
    Flags: 0x018 (PSH, ACK)
    Window: 17520
    [Calculated window size: 17520]
    [Window size scaling factor: -2 (no window scaling used)]
    Checksum: 0x1fbd [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
    [Timestamps]
    [SEQ/ACK analysis]
        [iRTT: 0.023265000 seconds]
        [Bytes in flight: 565]
        [Bytes sent since last PSH flag: 565]
    TCP payload (565 bytes)
    [Reassembled PDU in frame: 199]
    TCP segment data (565 bytes)
```

The sequence number of the TCP segment containing the HTTP POST command is 232129013 as seen in blue.

7. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)?

   At what time was each segment sent?

   When was the ACK for each segment received?

   Given the difference between when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments?

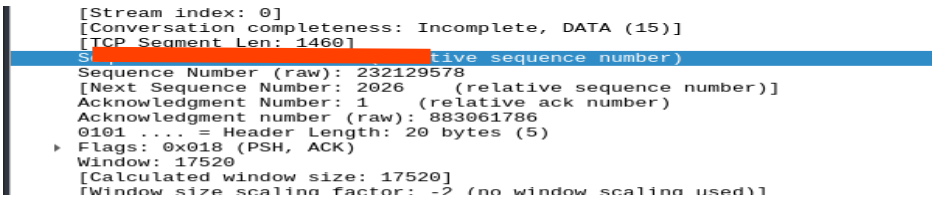
   What is the EstimatedRTT value (see Section 3.5.3, page 269 in text) after the receipt of each ACK? Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 270 for all subsequent segments.

Note: Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the "listing of captured packets" window that is being sent from the client to the gaia.cs.umass.edu server. Then select: Statistics->TCP Stream Graph->Round Trip Time Graph.

Frame	Segment number	Time segment sent	ACK recived	RTT	Estimated RTT
4	232129013	0,026477	0,053937	0,02746	0,02746
5	232129578	0,041737	0,077294	0,035557	0,02847
7	232131038	0,054026	0,124085	0,070059	0,03367
8	232132498	0,05469	0,169118	0,114428	0,04376
10	232133958	0,077405	0,217299	0,139894	0,05578
11	232135418	0,078157	0,267802	0,189645	0,07251

8. What is the length of each of the first six TCP segments?

Frame 4 is 565 bytes as seen in green in question 6 and the rest of the segments are 1460 bytes long as seen in red.



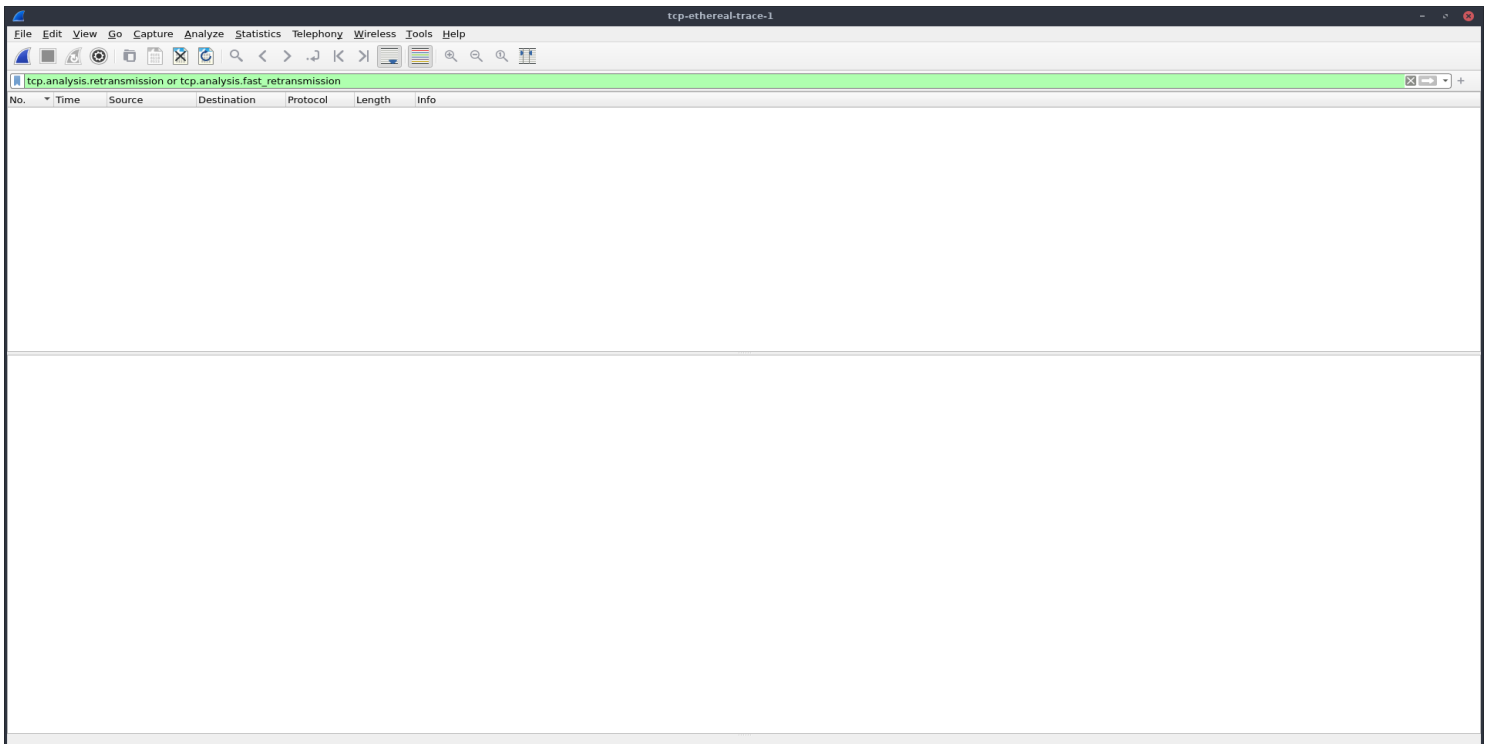
9. What is the minimum amount of available buffer space advertised at the receiver for the entire trace?

```
/home/viles222/Documents/wireshark/tcp-ethereal-trace-1 213 total packets, 202 shown

No.     Time           Source                Destination           Protocol Length Info
     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
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: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 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]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 883061785
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 232129013
    0111 .... = Header Length: 28 bytes (7)
    Flags: 0x012 (SYN, ACK)
    Window: 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
        TCP Option - Maximum segment size: 1460 bytes
            Kind: Maximum Segment Size (2)
            Length: 4
            MSS Value: 1460
        TCP Option - No-Operation (NOP)
        TCP Option - No-Operation (NOP)
        TCP Option - SACK permitted
    [Timestamps]
    [SEQ/ACK analysis]
```

The mimimum amount of available buffer space is found in the TCP SYN ACK in frame 2 where the window size is 5840 bytes.

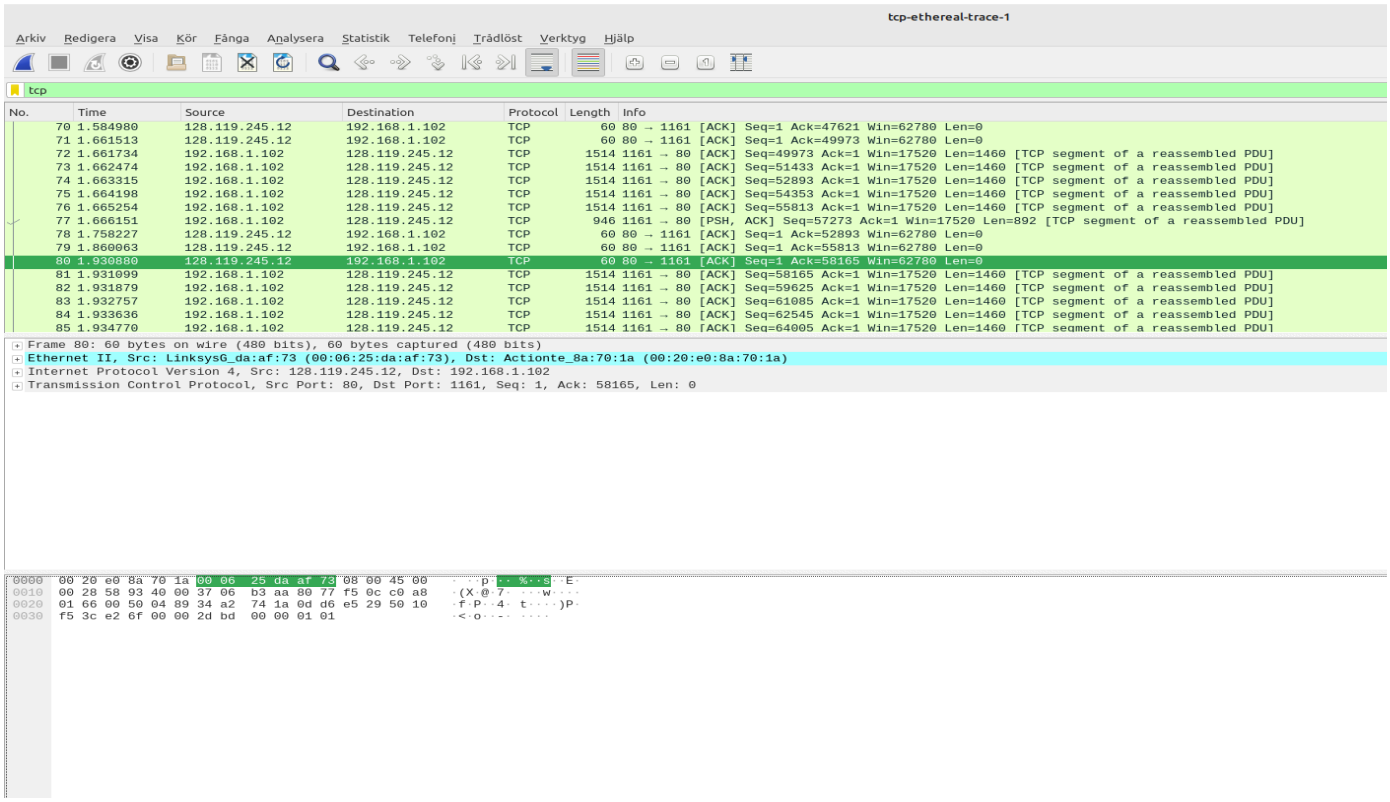
10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?



There are no retransmitted segment in the trace file. To check in the trace we would look for duplicated segment numbers and duplicated ACKs.

11. How much data does the receiver typically acknowledge in an ACK? Identify instances in the packet trace where the receiver acknowledges two segments instead of one. Hint: Examine the sequence and ACK numbers to detect this pattern (see Table 3.2 on page 278 in the text).

The reciver typically ACKs 1460 bytes. An instance where the reciver acks two segments is for example in frame 80 where two 1460 packets are ACK ed once.



12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.

Throughout = total sent bytes in tcp connection / time = 29036,773 bytes per second.

time = 0.026477 – 5.65111

total sent bytes == 164090

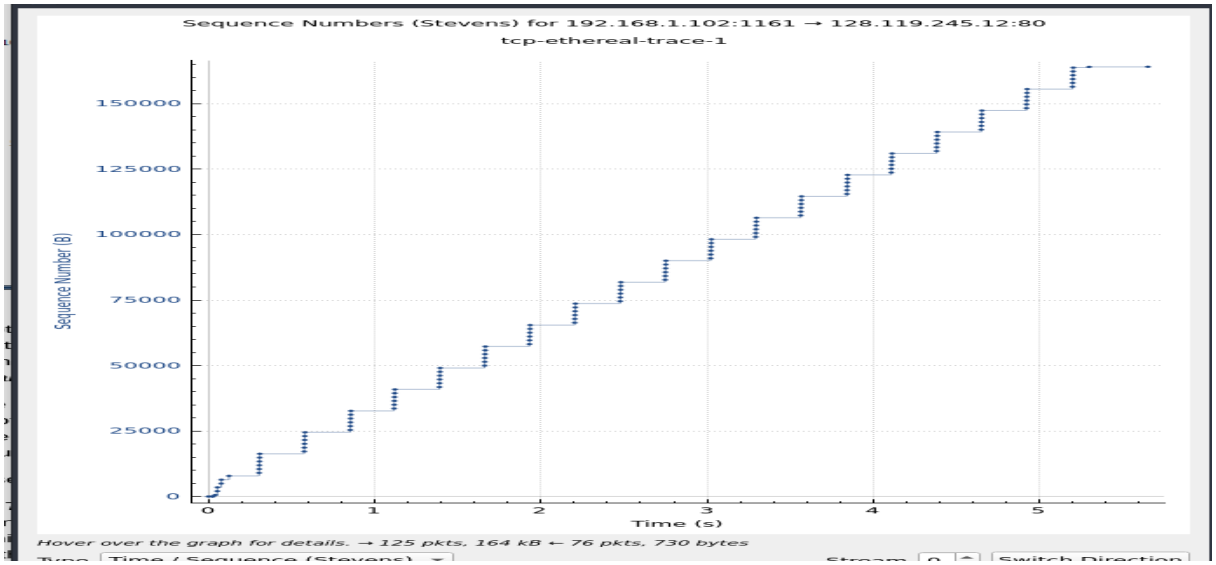
Task A: Now, based on questions 1-12, please write two paragraphs explaining and discussing your observations from the above questions. One paragraph should describe and discuss the connections at a high level. The second paragraph should discuss the impact of RTT estimates, packet losses, and interpreted packet loss events. Note that your answer may benefit from explaining and/or referring to some of your observations from the practice questions explicitly. Note that, similar to previous assignments, you are expected to convince us that you understand these aspects of TCP.

Disscussing the connections at a highlevel, the tcp connection begins in frame 4 and ends in frame 199. With a local computer transfering data to gaia.umass.edu through http port 80. After the syn ack is completed the tcp connection is realized and the client starts sending 560 and 1460 bytes size packes of data over to the server.

Disscussing the impact of RTT estimates, packet losses and interpreted packet loss events

Our RTT estimated did deviate by a significant bit from the real values. In the TCP connection there were no packet losses or packet loss events as there are no ack duplicates found at all in the trace.

13. Use the *Time-Sequence-Graph (Stevens)* plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Can you identify any indications of packet loss in this communication. What patterns in the sequence numbers suggest lost packets?



There are no indications of packet loss as if so there would be a pattern of repeated ACKs sent.

14. Explain the relationship between (i) the congestion window (`cwnd`), (ii) the receiver advertised window (`rwnd`), (iii) the number of unacknowledged bytes, and (iv) the effective window at the sender (i.e., the window effectively limiting the data transmission).

(i). The congestion window is variable the sender maintains for congestion control which is used to give the sender an idea of how much free buffer space is avaviable at the receiver and will dynamically adjust for network conditions. While the rwnd (ii) is the amount of buffer space avaviable at the reviver and is advertised in the tcp header. It is used to prevent the sender from sending too much data at once. (iii) The number of unacknowledged bytes is the amount of data the sender has sent but has not yet been acknowlaged for by the reciver. The sender will keep the amount unacknowlaged data less than the value of rwnd to prevent itself from overflowing the buffer of the reciver. (iv) The effective window at the sender is the smaller value of the congestion windon and the rwnd minus the unack:ed bytes and is the value limiting the transmission.

15. Is it **generally** possible to find the congestion window size (`cwnd`) and how it changes with time, from the captured trace files? If so, please explain how. If not, please explain when and when not. Motivate your answer and give examples.

It is now generally possible to find the cnwd value from a captured trace as it is a variable maintanied at the sender and is not transmitted explicitly in the tcp package headers. Yet it can be estimate its behaviour from observations as retransmissions, duplicate acknolagements or packet losses where the cwnd adjust and from which one can infer the cwnd indirectly.

16. What is the throughput of each of the connections in bps (bits per second)? What is the total bandwidth of the host on which the clients are running? Discuss the TCP fairness for this case.

Connection	Total transferred bytes	Duration (in seconds)	RTT (in milliseconds)	average throughput of connection
1	165095720	521	12	2535059,0403071
2	165842766	521	12	2546529,99616123
3	165458792	514	12	2575234,11673152
4	163235772	512	12	2550558,9375

17.What is the throughput of each of the connections in bps (bits per second)? What is the total bandwidth of the host on which the clients are running? Discuss the TCP fairness for this case.

Connection	Total transferred bytes	Duration (in seconds)	RTT (in milliseconds)	average throughput of connection in mega bits per second	Total bandwith of host in Mbits per sec
1	261319130	90	13	23,2250123870996	93,8862172519062
2	175995832	90	35	15,6379925140223	
3	151894552	90	68	13,4915443442732	
4	140388568	90	73	12,4688701830737	
5	108610702	90	49	9,64903126075803	
6	70644690	90	33	6,27722635033821	
7	65744938	90	135	5,83524162644921	
8	43212876	90	326	3,82728127006621	
9	39222524	90	322	3,4740173158256	

A tcp connection is considered fair if each connection gets and equal share of the bandwidth.

18. Discuss the TCP fairness for this case. How does it differ from the previous cases, and how is it affected by the use of BitTorrent?

Connection	Total transferred bytes	Duration (in seconds)	RTT (in milliseconds)
1	108851134	58	40
2	90435681	58	36
3	57971584	53	100
4	32000012	29	68
5	32557334	35	31
6	27199361	31	33
7	26329578	31	122
8	38834490	56	146
9	23571761	35	74
10	36252962	55	66

For this case it is harder to discuss tcp fairness as the RTT and the duration differs a lot for most of the connection. Connections 1 and 2, and 4 and 9 have similar durations and similar RTT but different amounts of transferred bytes hinting on TCP unfairness. This might be the cause of the use of bittorrents use of multiple TCP connections at once which might make the connections compete with each single connections tcp running at the same time.

For all of these questions you must take a closer look at the relationships between the characteristics of the different connections and discuss your findings in the context of the different experiments. You are expected to show that you understand the concept of TCP fairness and how the different scenarios may impact the throughput relationships that you observe and those that you may expect in general. To help the discussion you may for example want to create a scatter plot that show the estimated round trip time (RTT) and throughput against each other (for the different connections). You also want to carefully examine and discuss the above throughput equation and how it may apply to each scenario.