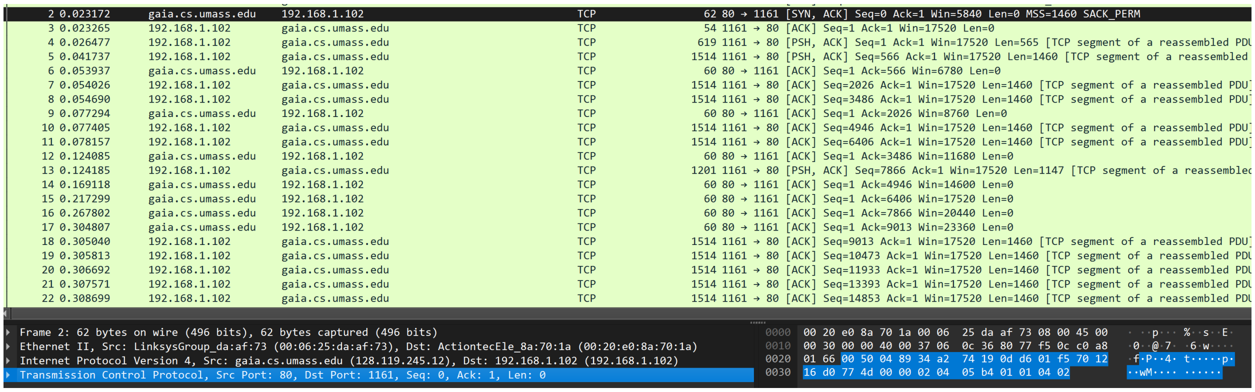
Lab04

Exercise 1

1.1) What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection? What are the IP address and TCP port numbers used by the client computer (source) that is transferring the file to gaia.cs.umass.edu?



```
 2 0.023172   gaia.cs.umass.edu   192.168.1.102       TCP     62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
 3 0.023265   192.168.1.102       gaia.cs.umass.edu   TCP     54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
 4 0.026477   192.168.1.102       gaia.cs.umass.edu   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       gaia.cs.umass.edu   TCP   1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled
 6 0.053937   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
 7 0.054026   192.168.1.102       gaia.cs.umass.edu   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       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 9 0.077294   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10 0.077405   192.168.1.102       gaia.cs.umass.edu   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       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
12 0.124085   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13 0.124185   192.168.1.102       gaia.cs.umass.edu   TCP   1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled
14 0.169118   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15 0.217299   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16 0.267802   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17 0.304807   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18 0.305040   192.168.1.102       gaia.cs.umass.edu   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       gaia.cs.umass.edu   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       gaia.cs.umass.edu   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       gaia.cs.umass.edu   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       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU
```

```
Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
Ethernet II, Src: LinksysGroup_da:af:73 (00:06:25:da:af:73), Dst: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a)
Internet Protocol Version 4, Src: gaia.cs.umass.edu (128.119.245.12), Dst: 192.168.1.102 (192.168.1.102)
Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0
```

```
0000  00 20 e0 8a 70 1a 00 06  25 da af 73 08 00 45 00   · ··p··· %··s··E·
0010  00 30 00 00 40 00 37 06  0c 36 80 77 f5 0c c0 a8   ·0··@·7· ·6·w····
0020  01 66 00 50 04 89 34 a2  74 19 0d d6 01 f5 70 12   ·f·P··4· t·····p·
0030  16 d0 77 4d 00 00 02 04  05 b4 01 01 04 02         ··wM···· ······
```

img

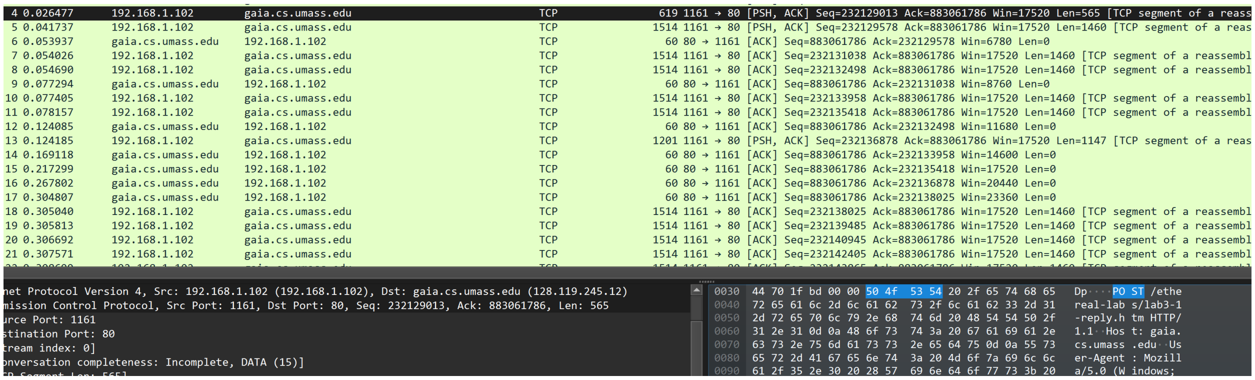
For gaia.cs.umass.edu:

- The IP address is: 128.119.245.12
- It sends and receives TCP segments on port 80

For the client it has:

- IP Address: 192.168.1.102
- port: 1161

1.2) What is the sequence number of the TCP segment containing the HTTP POST command?



```
 4 0.026477   192.168.1.102       gaia.cs.umass.edu   TCP    619 1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565 [TCP segment of a reass
 5 0.041737   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [PSH, ACK] Seq=232129578 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reas
 6 0.053937   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=883061786 Ack=232129578 Win=6780 Len=0
 7 0.054026   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232131038 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
 8 0.054690   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232132498 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
 9 0.077294   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=883061786 Ack=232131038 Win=8760 Len=0
10 0.077405   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232133958 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
11 0.078157   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232135418 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
12 0.124085   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=883061786 Ack=232132498 Win=11680 Len=0
13 0.124185   192.168.1.102       gaia.cs.umass.edu   TCP   1201 1161 → 80 [PSH, ACK] Seq=232136878 Ack=883061786 Win=17520 Len=1147 [TCP segment of a reas
14 0.169118   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=883061786 Ack=232133958 Win=14600 Len=0
15 0.217299   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=883061786 Ack=232135418 Win=17520 Len=0
16 0.267802   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=883061786 Ack=232136878 Win=20440 Len=0
17 0.304807   gaia.cs.umass.edu   192.168.1.102       TCP     60 80 → 1161 [ACK] Seq=883061786 Ack=232138025 Win=23360 Len=0
18 0.305040   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232138025 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
19 0.305813   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232139485 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
20 0.306692   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232140945 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
21 0.307571   192.168.1.102       gaia.cs.umass.edu   TCP   1514 1161 → 80 [ACK] Seq=232142405 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembl
```

```
net Protocol Version 4, Src: 192.168.1.102 (192.168.1.102), Dst: gaia.cs.umass.edu (128.119.245.12)
mission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 232129013, Ack: 883061786, Len: 565
urce Port: 1161
stination Port: 80
tream index: 0]
onversation completeness: Incomplete, DATA (15)]
```

```
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;
```

img

The sequence number containing the HTTP POST command is: **232129013**

1.3) Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection

1.3.a) What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST) <u> sent from the client to the webserver </u> (Do not consider the ACKs received from the server as part of these six segments)?

Packet	TCP Packet No (Sender to Client)	Sequence Number
1	4	232129013
2	5	232129578
3	7	232131038
4	8	232132498
5	10	232133958
6	11	232135418

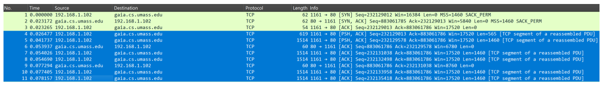
1.3.b) At what time was each segment sent? When was the ACK for each segment received? Given the difference between when each TCP segment was sent and when its acknowledgement was received, what is the RTT value for each of the six segments?

Packet	TCP Packet No (Sender to Client)	Sequence Number	Time Sent (s)	Acked Time (s)	RTT Value(s)
1	4	232129013	0.026477	0.053937	0.02746
2	5	232129578	0.041737	0.077294	0.035557
3	7	232131038	0.054026	0.124085	0.070059
4	8	232132498	0.05469	0.169118	0.114428
5	10	232133958	0.077405	0.217299	0.139894
6	11	232135418	0.078157	0.267802	0.189645

1.3.c) What is the EstimatedRTT value (see relevant parts of Section 3.5 or lecture slides) after receiving each ACK? Assume that the initial value of EstimatedRTT is equal to the measured RTT (SampleRTT) for the first segment and then is computed using the EstimatedRTT equation for all subsequent segments. Set alpha to 0.125

Packet	TCP Packet No (Sender to Client)	Sequence Number	Time Sent (s)	Acked Time (s)	RTT Value(s)	Estimated RTT(s)
1	4	232129013	0.026477	0.053937	0.02746	0.02746
2	5	232129578	0.041737	0.077294	0.035557	0.0284721
3	7	232131038	0.054026	0.124085	0.070059	0.0336705
4	8	232132498	0.05469	0.169118	0.114428	0.0437652
5	10	232133958	0.077405	0.217299	0.139894	0.0557813
6	11	232135418	0.078157	0.267802	0.189645	0.0725142

(d) What is the length of each of the first six TCP segments?

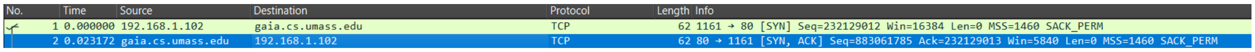


```
No.   Time     Source             Destination        Protocol  Length Info
    1 0.000000 192.168.1.102      gaia.cs.umass.edu  TCP         62 1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM
    2 0.023172 gaia.cs.umass.edu  192.168.1.102      TCP         62 80 → 1161 [SYN, ACK] Seq=883061785 Ack=232129013 Win=5840 Len=0 MSS=1460 SACK_PERM
    3 0.023265 192.168.1.102      gaia.cs.umass.edu  TCP         54 1161 → 80 [ACK] Seq=232129013 Ack=883061786 Win=17520 Len=0
    4 0.026477 192.168.1.102      gaia.cs.umass.edu  TCP        619 1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565 [TCP segment of a reassembled PDU]
    5 0.041737 192.168.1.102      gaia.cs.umass.edu  TCP       1514 1161 → 80 [PSH, ACK] Seq=232129578 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
    6 0.053937 gaia.cs.umass.edu  192.168.1.102      TCP         60 80 → 1161 [ACK] Seq=883061786 Ack=232129578 Win=6780 Len=0
    7 0.054026 192.168.1.102      gaia.cs.umass.edu  TCP       1514 1161 → 80 [ACK] Seq=232131038 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
    8 0.054690 192.168.1.102      gaia.cs.umass.edu  TCP       1514 1161 → 80 [ACK] Seq=232132498 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
    9 0.077294 gaia.cs.umass.edu  192.168.1.102      TCP         60 80 → 1161 [ACK] Seq=883061786 Ack=232131038 Win=8760 Len=0
   10 0.077405 192.168.1.102      gaia.cs.umass.edu  TCP       1514 1161 → 80 [ACK] Seq=232133958 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   11 0.078157 192.168.1.102      gaia.cs.umass.edu  TCP       1514 1161 → 80 [ACK] Seq=232135418 Ack=883061786 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
```

img

Packet	TCP Packet No (Sender to Client)	Packet Length
1	4	565
2	5	1460
3	7	1460
4	8	1460
5	10	1460
6	11	1460

1.4) What is the minimum amount of available buffer space advertised at the receiver for the entire trace? Does the lack of receiver buffer space ever throttle the sender?



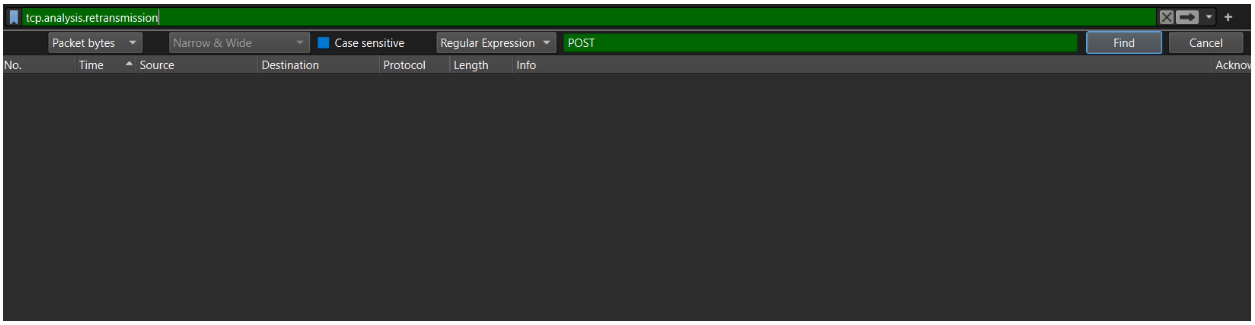
```
No.  Time     Source             Destination        Protocol  Length Info
  1 0.000000 192.168.1.102      gaia.cs.umass.edu  TCP         62 1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM
  2 0.023172 gaia.cs.umass.edu  192.168.1.102      TCP         62 80 → 1161 [SYN, ACK] Seq=883061785 Ack=232129013 Win=5840 Len=0 MSS=1460 SACK_PERM
```

img

Minimum buffer space advertised in the entire trace is: 5840

The receiver buffer space does not bottleneck (throttle) the sender because the window size does not decrease. It increases from 5840 and stays at 62780.

1.5) Are there any retransmitted segments in the trace file? To answer this question, what did you check for (in the trace)?



tcp.analysis.retransmission

Packet bytes   Narrow & Wide   ■ Case sensitive   Regular Expression   POST

img

There are no retransmitted segments in the trace file as seen by the tcp.analysis.retransmission filter.

1.6) How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (recall the discussion about delayed acks from the lecture notes or Section 3.5 of the text)?

The receiver typically acknowledges **1460** bytes of data in an ack.

The receiver starts to use delayed cumulative ack and seems to begin from packet No. 61 (**3920** bytes of data) onwards and there are many other cases of the receiver doing this.

Between No. 87-89 we can see the receiver (gaia.cs.umass.edu) using a cumulative ack to acknowledge the 2 packets in between 81-86 because from the calculation below:

Packet No. 87 has seqnum $$232190097$$ and Packet No. 88 has seqno: $$232193017$$

The difference between them: $$(232193017 - 232190097) = 2920 = 2*1460$$ bytes of data acked

Which means the receiver is using a cumulative ack to acknowledge packet No. 81 and 82. And the other following packets

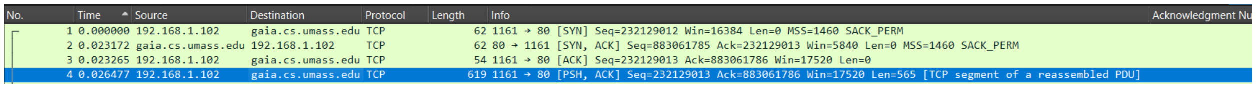
The receiver is doing this because it waits up to 500ms for the next segment. If it arrives it sends a cumulative ack. Otherwise, it just sends the ack of that newly receive message. This is known as a delayed ack

1.7) What is the TCP connection's throughput (bytes transferred per unit of time during the connection)? Explain how you calculated this value

Network Throughput is defined as:

$$ Throughput = { FileSize \over TimeTakenFileTransfer} $$

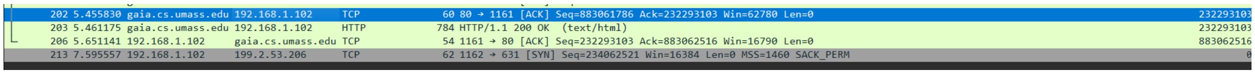
To get the actual time taken to transfer the file, we need to exclude the TCP setup and teardown phase times(syn and fin).



No.	Time	Source	Destination	Protocol	Length	Info	Acknowledgment Nu
1	0.000000	192.168.1.102	gaia.cs.umass.edu	TCP	62	1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM	
2	0.023172	gaia.cs.umass.edu	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=883061785 Ack=232129013 Win=5840 Len=0 MSS=1460 SACK_PERM	
3	0.023265	192.168.1.102	gaia.cs.umass.edu	TCP	54	1161 → 80 [ACK] Seq=232129013 Ack=883061786 Win=17520 Len=0	
4	0.026477	192.168.1.102	gaia.cs.umass.edu	TCP	619	1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565 [TCP segment of a reassembled PDU]	

img

We start at packet No. 4. So start time = 0.02647



202	5.455830	gaia.cs.umass.edu	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=883061786 Ack=232293103 Win=62780 Len=0	232293103
203	5.461175	gaia.cs.umass.edu	192.168.1.102	HTTP	784	HTTP/1.1 200 OK (text/html)	232293103
206	5.651141	192.168.1.102	gaia.cs.umass.edu	TCP	54	1161 → 80 [ACK] Seq=232293103 Ack=883062516 Win=16790 Len=0	883062516
213	7.595557	192.168.1.102	199.2.53.206	TCP	62	1162 → 631 [SYN] Seq=234062521 Win=16384 Len=0 MSS=1460 SACK_PERM	0

img

We end at packet No.202 because we need to ack the last data packet. So the end time = 5.455830

So the total time taken to transfer the file is: $$5.455830s - 0.02647s = 5.42936s $$

The FileSize can be found by taking the difference between Packet No. 202's acknowledgement number and Packet No 4 's sequence number $$ 232293103 - 232129013 = 164,090 \text{ bytes} $$

Therefore the Throughput is: $${164,090 \over 5.42936} = 30,222.715 \text{ bytes / sec} = 30.222 \text{ KB / s}$$

Exercise 2

2.1) What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and server?

The sequence number of the TCP SYN Segment that starts the TCP connection is: **2818463618**

2.2) What is the sequence number of the SYNACK segment sent by the server to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did the server determine that value?

The replied sequence number of the TCP SYNACK segment is: **1247095790**

The value of the Acknowledgement field in the SYNACK Segment is: **2818463619**

The server determined this value by incrementing the client's seequence number by one: $2818463618 + 1 = 2818463619$

2.3) What is the sequence number of the ACK segment sent by the client computer in response to the SYNACK? What is the value of the Acknowledgment field in this ACK segment? Does this segment contain any data?

Sequence number of the ACK segment sent by the client: $2818463619$

Acknowledgment number in the ACK segment is: 12470957910

The segment does not contain any data as the sequence number in packet No. 298 is the same in packet No.297

2.4) Who has done the active close? Is it the client or the server? How you have determined this? What type of closure has been performed? 3 Segment (FIN/FINACK/ACK), 4 Segment (FIN/ACK/FIN/ACK) or Simultaneous close?

A simultaneous close has been conducted.

Both the client and server has initiated the active close by sending the (FIN,ACK) segment. This is becuase both the client and server sent a (FIN,ACK) segment without receiving a (FIN) segment first.

More so the sequence number in Packet No.304 is the same for the acknowledgement number in Packet No.305 instead of (seqnum(No304) + 1).

Lastly, the both increment the sequence numbers by 1 and acknowledge the FIN segment to indicate a closed connection.

Therefore a Simultaneous has been performed.

2.5) How many data bytes have been transferred from the client to the server and from the server to the client during the whole duration of the connection? What relationship does this have with the Initial Sequence Number and the final ACK received from the other side?

Client:

$$ (1) \text{Client inital sequence number after setup (Exclude SYN)} = 2818463619$$

$$(2) \text{Client final sequence number after sending data (Exlcude FIN)} = 2818463652$$

$$ \text{Bytes Sent By Client} = (2) - (1) = 2818463652 - 2818463619 = 33 \text{ bytes} $$ Server:

$$ (1) \text{Server inital sequence number after setup (Exclude SYN)} = 1247095791 $$

$$(2) \text{Server final sequence number after sending data (Exlcude FIN)} = 1247095831$$

$$ \text{Bytes Sent By Server} = (2) - (1) = 1247095831 - 1247095791 = 40 \text{ bytes} $$

Initially, during the connection setup the relationship is that each side has it's own sequence number and we increment the sequence number by one and assign it to the acknowledgement number in the SynAck.

Then the relationship is that we increment the sequence number with the length of the TCP data sent. This will be put in the acknowledgement number in the response packet and repeats until we reach the connection tear down phase.

During the teardown phase, we also increment the sequence number by one when sending a response to the FIN segment.

At sender: $$\text{Initial SeqNum} = \text{ackNumber of received packet}$$ At receiver: $$\text{ackNumber} = \text{SeqNum of received packet} + \text{lengthTCPSegment}$$

And if a SYN or FIN segment is received, then increment the sequence number by 1 and assign it as the acknowledgement number