CS646-Computer Communication Networks
Lab Assignment-2
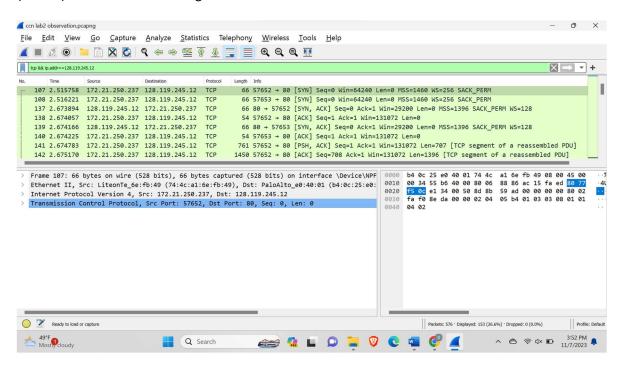
Name: Sammed Jain Dharmapal Prakash
Student ID: 02071953

1. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?
Answer:
The IP address 172.27.237.191 and the Source port number 57652 were used by your client computer (source) to transfer the file to gaia.cs.umass.edu.

2. 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?
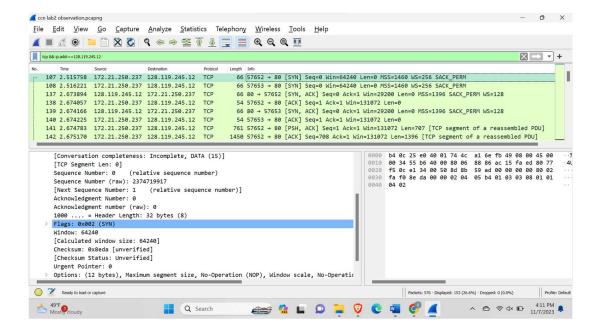**Answer:**
The sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu is 0.
The flag [SYN] identifies the segment as a SYN segment.

3. 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?

Answer:

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

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

gaia.cs.umass.edu determined the value by adding 1 to the initial sequence number.

The flag [SYN,ACK] identifies the segment as a SYNACK segment.

4. What is the sequence number of the TCP segment containing the HTTP POST command?
Answer:
The sequence number of the TCP segment containing the HTTP POST command is 152872.

5. Identify the 3 TCP ACKs from the sever, and find their corresponding TCP data segments from the client. Explain how did you find these 3 TCP data segments. At what time was each TCP data segment sent by the client? When was each TCP ACK received? Given the difference, what are the RTT values for these 3 TCP segments?
Answer:
case 1
Let us consider the ACK in frame 239. The acknowledgement number is 20252. The frame 170 shows one of the TCP data segments from the client where seq=18856 and len=1396. If we add seq and len of this TCP data segment we get 20252 which is equal to the ack number in the frame 239. Therefore, frame 170 refers to the corresponding TCP data segments from the client for the acknowledgement in the frame 239.
The time at which TCP data segment sent by the client:2.721888.
The TCP ACK was received at:3.077449
RTT value: 3.077449-2.721888=0.355561.

Case 2

Let us consider the ACK in frame 244. The acknowledgement number is 27232. The frame 175 shows one of the TCP data segments from the client where seq=25836 and len=1396. If we add seq and len of this TCP data segment we get 27232 which is equal to the ack number in the frame 244. Therefore, frame 175 refers to the corresponding TCP data segments from the client for the acknowledgement in the frame 244.

The time at which TCP data segment was sent by the client:2.721888.

The TCP ACK was received at:3.082761.

RTT value: 3.082761 -2.721888=0.360873.

Case 3

Let us consider the ACK in frame 266. The acknowledgement number is 39796. Frame 184 shows one of the TCP data segments from the client where seq=38400 and len=1396. If we add seq and len of this TCP data segment we get 39796 which is equal to the ack number in the frame 266. Therefore, frame 184 refers to the corresponding TCP data segments from the client for the acknowledgement in frame 266.
The time at which TCP data segment was sent by the client:2.721888.
The TCP ACK was received at:3.080297.
RTT value: 3. 080297-2.721888=0.358409.

6. Among the 3 TCP ACKs you identified in the previous question, what are the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?

Answer:

File   Edit   View   Go   Capture   Analyze   Statistics   Telephony   Wireless   Tools   Help

tcp && ip.addr==128.119.245.12

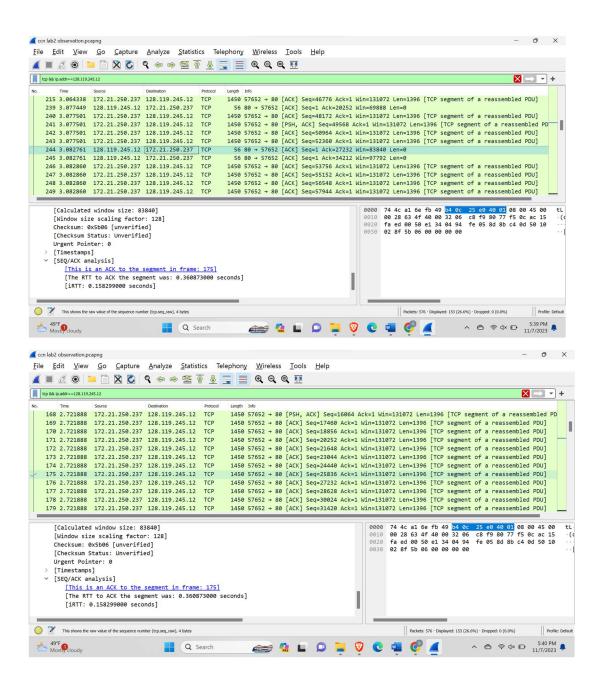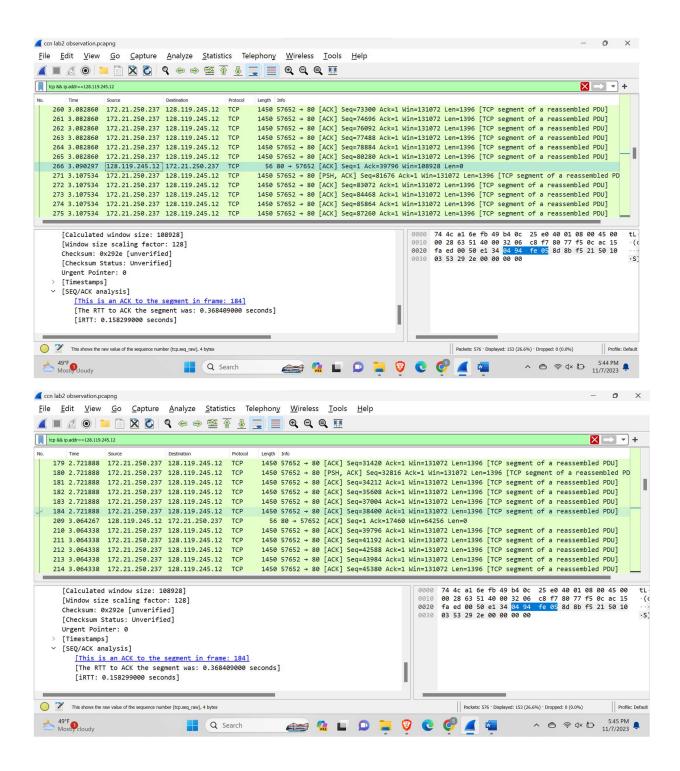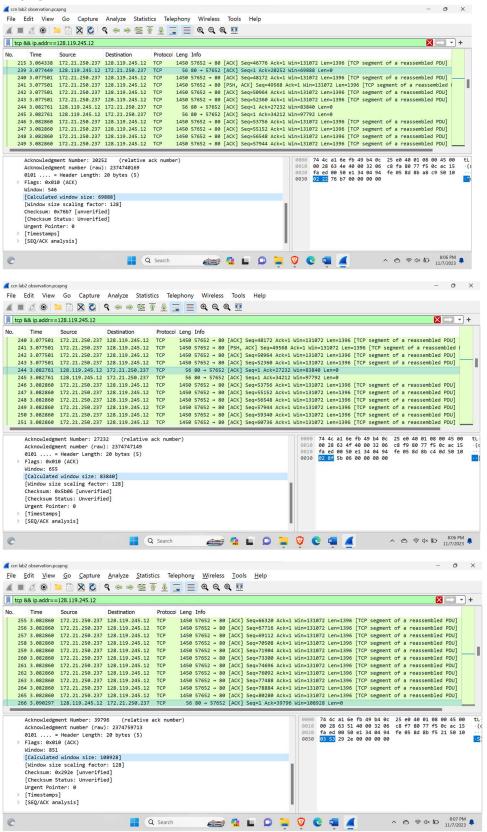| No.	Time	Source	Destination	Protocol	Leng	Info
215	3.064338	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=46776 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
239	3.077449	128.119.245.12	172.21.250.237	TCP	56	80 → 57652 [ACK] Seq=1 Ack=20252 Win=69888 Len=0
240	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=48172 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
241	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [PSH, ACK] Seq=49568 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled
242	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=50964 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
243	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=52360 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
244	3.082761	128.119.245.12	172.21.250.237	TCP	56	80 → 57652 [ACK] Seq=1 Ack=27232 Win=83840 Len=0
245	3.082761	128.119.245.12	172.21.250.237	TCP	56	80 → 57652 [ACK] Seq=1 Ack=34212 Win=97792 Len=0
246	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=53756 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
247	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=55152 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
248	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=56548 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
249	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=57944 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]

Acknowledgment Number: 20252   (relative ack number)
Acknowledgment number (raw): 2374740169
0101 .... = Header Length: 20 bytes (5)
> Flags: 0x010 (ACK)
Window: 546
[Calculated window size: 69888]
[Window size scaling factor: 128]
Checksum: 0x76b7 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
> [Timestamps]
> [SEQ/ACK analysis]

```
0000  74 4c a1 6e fb 49 b4 0c  25 e0 40 01 08 00 45 00   tL
0010  00 28 63 4e 40 00 32 06  c8 fa 80 77 f5 0c ac 15   ·(
0020  fa ed 00 50 e1 34 04 94  fe 05 8d 8b a8 c9 50 10
0030  02 22 76 b7 00 00 00 00                            ·"
```

8:06 PM 11/7/2023

File   Edit   View   Go   Capture   Analyze   Statistics   Telephony   Wireless   Tools   Help

tcp && ip.addr==128.119.245.12

No.	Time	Source	Destination	Protocol	Leng	Info
240	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=48172 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
241	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [PSH, ACK] Seq=49568 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled
242	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=50964 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
243	3.077501	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=52360 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
244	3.082761	128.119.245.12	172.21.250.237	TCP	56	80 → 57652 [ACK] Seq=1 Ack=27232 Win=83840 Len=0
245	3.082761	128.119.245.12	172.21.250.237	TCP	56	80 → 57652 [ACK] Seq=1 Ack=34212 Win=97792 Len=0
246	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=53756 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
247	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=55152 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
248	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=56548 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
249	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=57944 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
250	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=59340 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
251	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=60736 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]

Acknowledgment Number: 27232   (relative ack number)
Acknowledgment number (raw): 2374747149
0101 .... = Header Length: 20 bytes (5)
> Flags: 0x010 (ACK)
Window: 655
[Calculated window size: 83840]
[Window size scaling factor: 128]
Checksum: 0x5b06 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
> [Timestamps]
> [SEQ/ACK analysis]

```
0000  74 4c a1 6e fb 49 b4 0c  25 e0 40 01 08 00 45 00   tL
0010  00 28 63 4f 40 00 32 06  c8 f9 80 77 f5 0c ac 15   ·(
0020  fa ed 00 50 e1 34 04 94  fe 05 8d 8b c4 0d 50 10
0030  02 8f 5b 06 00 00 00 00                            ·
```

8:06 PM 11/7/2023

File   Edit   View   Go   Capture   Analyze   Statistics   Telephony   Wireless   Tools   Help

tcp && ip.addr==128.119.245.12

No.	Time	Source	Destination	Protocol	Leng	Info
255	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=66320 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
256	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=67716 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
257	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=69112 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
258	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=70508 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
259	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=71904 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
260	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=73300 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
261	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=74696 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
262	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=76092 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
263	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=77488 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
264	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=78884 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
265	3.082860	172.21.250.237	128.119.245.12	TCP	1450	57652 → 80 [ACK] Seq=80280 Ack=1 Win=131072 Len=1396 [TCP segment of a reassembled PDU]
266	3.090297	128.119.245.12	172.21.250.237	TCP	56	80 → 57652 [ACK] Seq=1 Ack=39796 Win=108928 Len=0

Acknowledgment Number: 39796   (relative ack number)
Acknowledgment number (raw): 2374759713
0101 .... = Header Length: 20 bytes (5)
> Flags: 0x010 (ACK)
Window: 851
[Calculated window size: 108928]
[Window size scaling factor: 128]
Checksum: 0x292e [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
> [Timestamps]
> [SEQ/ACK analysis]

```
0000  74 4c a1 6e fb 49 b4 0c  25 e0 40 01 08 00 45 00   tL
0010  00 28 63 51 40 00 32 06  c8 f7 80 77 f5 0c ac 15   ·(
0020  fa ed 00 50 e1 34 04 94  fe 05 8d 8b f5 21 50 10
0030  03 53 29 2e 00 00 00 00                            ·S
```

8:07 PM 11/7/2023

The minimum amount of available buffer space advertised at the received for the entire trace are 69888. Lack of receiver buffer size will not throttle the sender because the file size that we are sending is very small compared to the receiver's capacity.

7. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.
Answer:
To calculate the throughput (in bytes transferred per unit time) for a TCP connection, we need to consider several factors and use the following formula.

Throughput (bytes per second) = TCP window size / Round-trip time (RTT)

TCP Window Size: The TCP window size is the maximum amount of data that can be sent from one end of the connection to the other without receiving an acknowledgment. It's determined during the TCP handshake and can vary based on the configuration of the sender and receiver.

Round-Trip Time (RTT): The Round-Trip Time is the time it takes for a packet to travel from the sender to the receiver and back. It includes both the propagation delay (time taken for the data to travel over the physical medium) and the processing delay at both ends.

So, to calculate the throughput, we need to know the TCP window size and the RTT. Once we have these values, we can use the formula above to determine the throughput in bytes per second.

Calculation:

The time stamp of the initiation of the handshake this TCP: 2.515758
The time stamp of the last acknowledgement of this TCP: 4.271051
Seq. of the last acknowledgement of this TCP: 153029
Throughput=153029/ (4.271051-2.515758)
Throughput=87181.4563 bytes/sec=85.1447 kbps

8. 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 where is TCP's slow start phase and congestion avoidance phase? Can you identify any congestion?
Answer:

slow start phase:
The slow start phase, which is indicated by the red rectangle starts from 0.04s to 0.26s. During the slow start, the sender (our System) is exponentially increasing the number of packets, sending more and more data. Initially at 0.04s, the number of packets sent is the number of packets sent are 9. At 0.10 seconds, the number of packets sent is 19. At 0.26 s, the number of packets sent is 38. Therefore, we can see that each time the packets being sent are (doubling) increasing exponentially which indicates a slow start phase.

Congestion avoidance phase:

The Congestion avoidance phase, which is indicated by the red rectangle starts from 0.26s to 0.33s. During Congestion avoidance phase, the sequence number versus time plot shows a steadier, linear increase. the sender (our system) is gradually increasing the number of packets sent, probing for available bandwidth without causing congestion. At 0.26s, the number of packets sent is 38. At 0. 33s, the number of packets sent is 40. Therefore, we can see that unlike the slow start phase, the number of packets sent is (gradually increasing) increasing linearly which indicates the congestion avoidance phase.

Congestions:

Sudden drops or discontinuities in the sequence number plot, as well as the appearance of duplicate acknowledgments (indicating retransmissions), are signs of congestion. There are no such scenarios happening this TCP handshake. Therefore, there is no congestion.