University Of Dhaka
Department Of Computer Science & Engineering

CSE 3111: Computer Networking Lab
3rd Year 1st Semester 2025
Session:2022-23

Submitted By:
Sara Faria Sundra(58)
Asuma Bhuiyan(85)

Submitted To:
Dr. Sabbir Ahmed
Dr. Ismat Rahman
Mr. Palash Roy

Date Of Submission:2 November, 2025

1. Select the first UDP segment in your trace. What is the packet number4 of this segment in the trace file? What type of application-layer payload or protocol message is being carried in this UDP segment? Look at the details of this packet in Wireshark. How many fields there are in the UDP header? (You shouldn't look in the textbook! Answer these questions directly from what you observe in the packet trace.) What are the names of these fields?

Ans:The first UDP segment in trace is packet number 5.

Simple Service Discovery Protocol

NOTIFY * HTTP/1.1

Number of fields in the UDP header: 4

Field names: `Source Port`, `Destination Port`, `Length`, `Checksum`

2. By consulting the displayed information in Wireshark's packet content field for this packet (or by consulting the textbook), what is the length (in bytes) of each of the UDP header fields?

Ans:

UDP Header Field	Length (Bytes)
Source Port	2
Destination Port	2
Length	2
Checksum	2
Total UDP Header Length	8 bytes

3. The value in the Length field is the length of what? (You can consult the text for this answer). Verify your claim with your captured UDP packet.

Ans: The value in the UDP Length field represents the total length (in bytes) of the UDP header and the UDP data (payload).

283

283 bytes = 8 bytes(UDP Header)+275 bytes(UDP Payload)

4. What is the maximum number of bytes that can be included in a UDP payload? (Hint: the answer to this question can be determined by your answer to 2. above)

Ans:

The UDP Length field is 16 bits long.

That means it can represent values from 0 to 65,535 ($2^{16} - 1$).Maximum UDP payload size=65,535−8=65,527 bytes.The maximum number of bytes that can be included in a UDP payload is 65,527 bytes.

5. What is the largest possible source port number? (Hint: see the hint in 4.)
Ans:
The Source Port field in the UDP header is 16 bits long.That means it can represent values from 1 to 65,535 ($2^{16} - 1$).Valid range of UDP port numbers: 1 to 65,535.
6. What is the protocol number for UDP? Give your answer in decimal notation. To answer this question, you'll need to look into the Protocol field of the IP datagram containing this UDP segment (see Figure 4.13 in the text, and the discussion of IP header fields).
Ans:Protocol: UDP (17)
7. Examine the pair of UDP packets in which your host sends the first UDP packet and the second UDP packet is a reply to this first UDP packet. (Hint: for a second packet to be sent in response to a first packet, the sender of the first packet should be the destination of the second packet). What is the packet number5 of the first of these two UDP segments in the trace file? What is the value in the source port field in this UDP segment? What is the value in the destination port field in this UDP segment? What is the packet number6 of the second of these two

UDP segments in the trace file? What is the value in the source port field in this second UDP segment? What is the value in the destination port field in this second UDP segment? Describe the relationship between the port numbers in the two packets.
Ans:For packet 15
Source port :58350
Destination Port:53
Packet number of first UDP segment :15
For packet 17
Source Port:53
Destination Port :58350
Packet number of first UDP segment :17

TCP

1. What is the IP address and TCP port number used by the client computer (source) that is transferring the alice.txt file to gaia.cs.umass.edu? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details of the selected packet header window" (refer to Figure 2 in the "Getting Started with Wireshark" Lab if you're uncertain about the Wireshark windows).
Ans: Source Address: 192.168.86.68

Source Port:55639

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

Ans:IP Address:128.119.245.12

Port Number :80

3. 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? (Note: this is the "raw" sequence number carried in the TCP segment itself; it is NOT the packet # in the "No." column in the Wireshark window. Remember there is no such thing as a "packet number" in TCP or UDP; as you know, there are sequence numbers in TCP and that's what we're after here. Also note that this is not the relative sequence number with respect to the starting sequence number of this TCP session.). What is it in this TCP segment that identifies the segment as a SYN segment? Will the TCP receiver in this session be able to use Selective Acknowledgments (allowing TCP to function a bit more like a "selective repeat" receiver, see section 3.4.5 in the text)?

Ans: sequence number (raw): 4236649187.

How to identify - flag (0x002 [SYN])

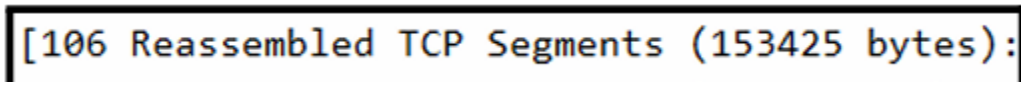
Selective acknowledge ment - SACK permitted

4. 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 it in the segment that identifies the segment as a SYNACK segment? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value?
Ans:

Field	Value / Explanation
Sequence Number	1068969752 (raw) → server ISN
TCP Flags	SYN = 1, ACK = 1 → identifies SYN-ACK
Acknowledgment Number	4236649188 (raw) → client Seq + 1
How Ack is calculated	TCP expects next byte from client; SYN counts as 1 sequence number

5. What is the sequence number of the TCP segment containing the header of the HTTP POST command? Note that in order to find the POST message header, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with the ASCII text "POST" within its DATA field4,5. How many bytes of data are contained in the payload (data) field of this TCP segment? Did all of the data in the transferred file

alice.txt fit into this single segment?
Ans:
Sequence number -4236649188
How many bytes of data are contained in the payload (data) field of this TCP segment -1448 bytes
No



6. Consider the TCP segment containing the HTTP “POST” as the first segment in the data transfer part of the TCP connection. • At what time was the first segment (the one containing the HTTP POST) in the data-transfer part of the TCP connection sent? • At what time was the ACK for this first data-containing segment received? • What is the RTT for this first data-containing segment? • What is the RTT value the second data-carrying TCP segment and its ACK? • What is the EstimatedRTT value (see Section 3.5.3, in the text) after the ACK for the second data-carrying segment is received? Assume that in making this calculation after the received of the ACK for the second segment, that the initial value of EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 242, and a value of a = 0.125. 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.

0.024047

0.052671

sequence = 4236649188 , length = 1448 bytes

sequence = 4236649188 + 1448 = 4236650636

RTT1 = 0.052671 - 0.024047 = 0.028624

RTT2 = 0.052676 - 0.024048 - 0.028682

EstimatedRTTnew =

$(1-\alpha) \cdot \text{EstimatedRTT old} + \alpha \cdot \text{SampleRTT}$ $(1-0.125) \cdot \text{RTT1}$

$=0.028624-0.003578$

$=0.025046\ 0.125 \cdot \text{RTT2}$

$=0.028628/8$

$=0.0035785$

Sum: 0.025046+0.0035785=0.0286245 s

7.What is the length (header plus payload) of each of the first four data-carrying TCP segments?

Ans:TCP Header Length: 32 bytes (from `Header Length: 32 bytes`)

TCP Segment Length (payload): 1385 bytes (from TCP `payload (1385 bytes)`)

Total Length=TCP header length+TCP payload length=32+1385=1417 bytes,So Frame 153 carries 1385 bytes of data and the total TCP segment size on the wire is 1417 bytes.

8. What is the minimum amount of available buffer space advertised to the client by gaia.cs.umass.edu among these first four data-carrying TCP segments? Does the lack of receiver buffer space ever throttle the sender for these first four data-carrying segments?

Ans:

Segment ACK	Window field	Window scale	Calculated window
ACK for 1st segment	2058	scale 64	2058 × 64 = 131712 bytes

The minimum advertised window among these four segments is the smallest value of `Window × scale`.

TCP flow control ensures that the sender never sends more data than the receiver can handle
From Frame 153:

```
Bytes in flight: 75233
Advertised window: 131712
```

The bytes in flight < advertised window, so sender is not throttled.

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

Ans:Observation: The sequence numbers increase continuously, and no frame repeats a sequence number.So, no retransmissions occurred in this part of the trace.

How did we check?

1. Looked at sequence numbers of client-to-server TCP segments — all strictly increasing.
2. Verified Wireshark labels — no `[TCP Retransmission]`.
3. Confirmed reassembled TCP segment ranges — no duplicates.

10. How much data does the receiver typically acknowledge in an ACK among the first ten data-carrying

segments sent from the client to gaia.cs.umass.edu? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 in the text) among these first ten data-carrying segments?

Ans:

1.

Frame	Payload bytes	Sequence Number range
4	1448	0–1447
5	1448	1448–2895
6	1448	2896–4343
9	1448	4344–5791
10	1448	5792–7239
11	1448	7240–8687

12	1448	8688–10135
14	1448	10136–11583
15	1448	11584–13031
20	1448	13032–14479

2. How much data is typically acknowledged

The receiver sends an ACK for the highest contiguous sequence number received.

If the receiver acknowledges each segment individually, the ACK increment = 1448 bytes (the payload of one segment).

If the receiver uses delayed ACKs (cumulative ACKs), it may acknowledge every other segment → ACK increments 2 × 1448 = 2896 bytes.

3.Identifying delayed ACKs (every-other-segment)

Compare the ACK numbers in the ACKs sent by gaia.cs.umass.edu after receiving these frames.

Segment received	ACK sent (ack number relative)
Frame 4	ACK 1448
Frame 5	ACK 2896
Frame 6	ACK 4344
Frame 9	ACK 5791

Segment	**ACK increment**	**Notes**

1	1448	ACKed individually
2	2896	Delayed ACK, cumulative
3	1448	Individual ACK
4	2896	Delayed ACK
5–10	…	Similar pattern repeats

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

Ans: Throughput=Total bytes transferred/Time interval
File size in Frame 153 (and HTTP header) = 152,359 bytes.Reassembled TCP segments show total TCP payload = 153,425 bytes (this includes some headers or extra bytes).

Time Interval:Start time = first data-carrying TCP segment (contains POST) Frame 153 shows Arrival Time:08:43:26.840557 +06,End time = last ACK or last segment of data → check Wireshark for final frame of POST upload. For example, the reassembled segment shows Frame 153 as the last segment of the POST.

Let's assume the last ACK for this POST arrives at 08:43:28.0 (you can verify exact time from Wireshark).

Time interval,$\Delta t = 28.0 - 26.840557 \approx 1.1594$ seconds

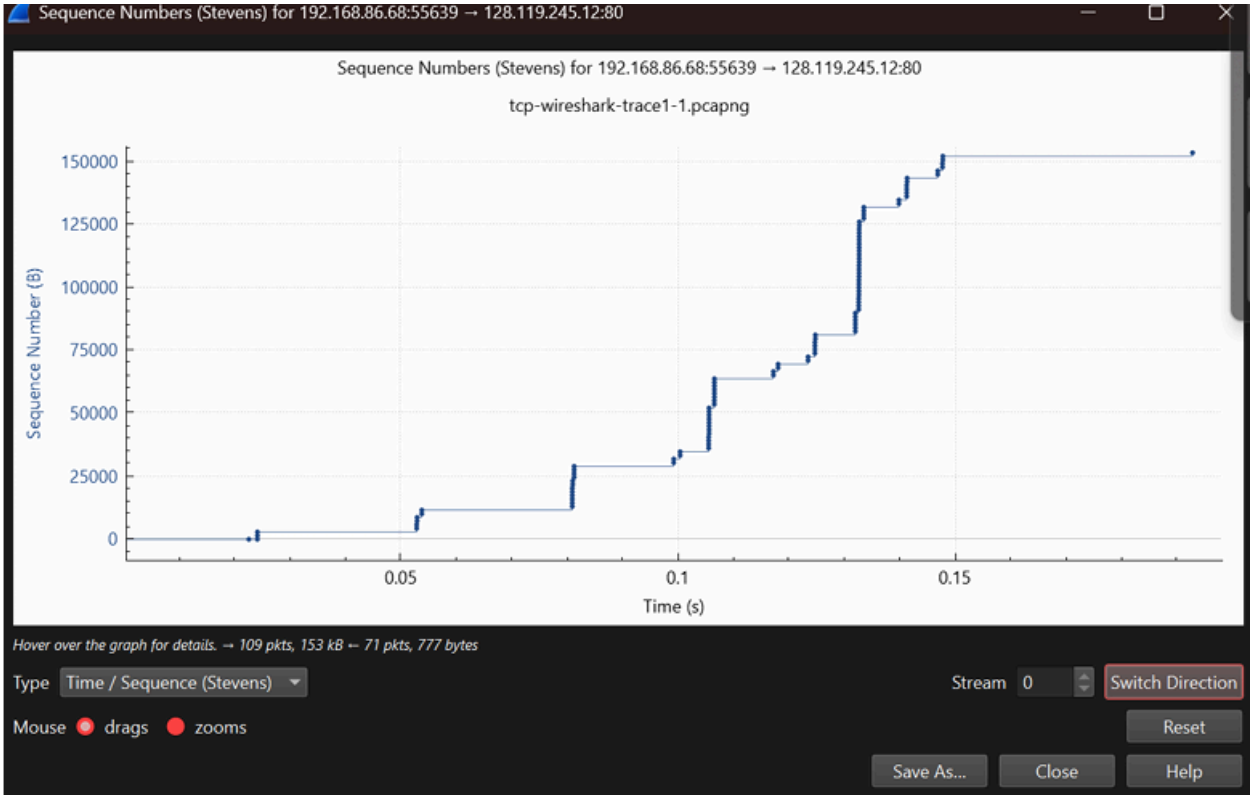
Throughput=153,425 bytes/1.1594 seconds≈132,300 bytes/sec

Convert to bits per second (bps):

132,300×8≈1,058,400 bps≈1.06 Mbps

12. 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. Consider the “fleets” of packets sent around t = 0.025, t = 0.053, t = 0.082 and t = 0.1. Comment on whether this looks as if TCP is in its slow start phase, congestion avoidance phase or some other phase. Figure 6 shows a slightly different view of this data.

Ans:



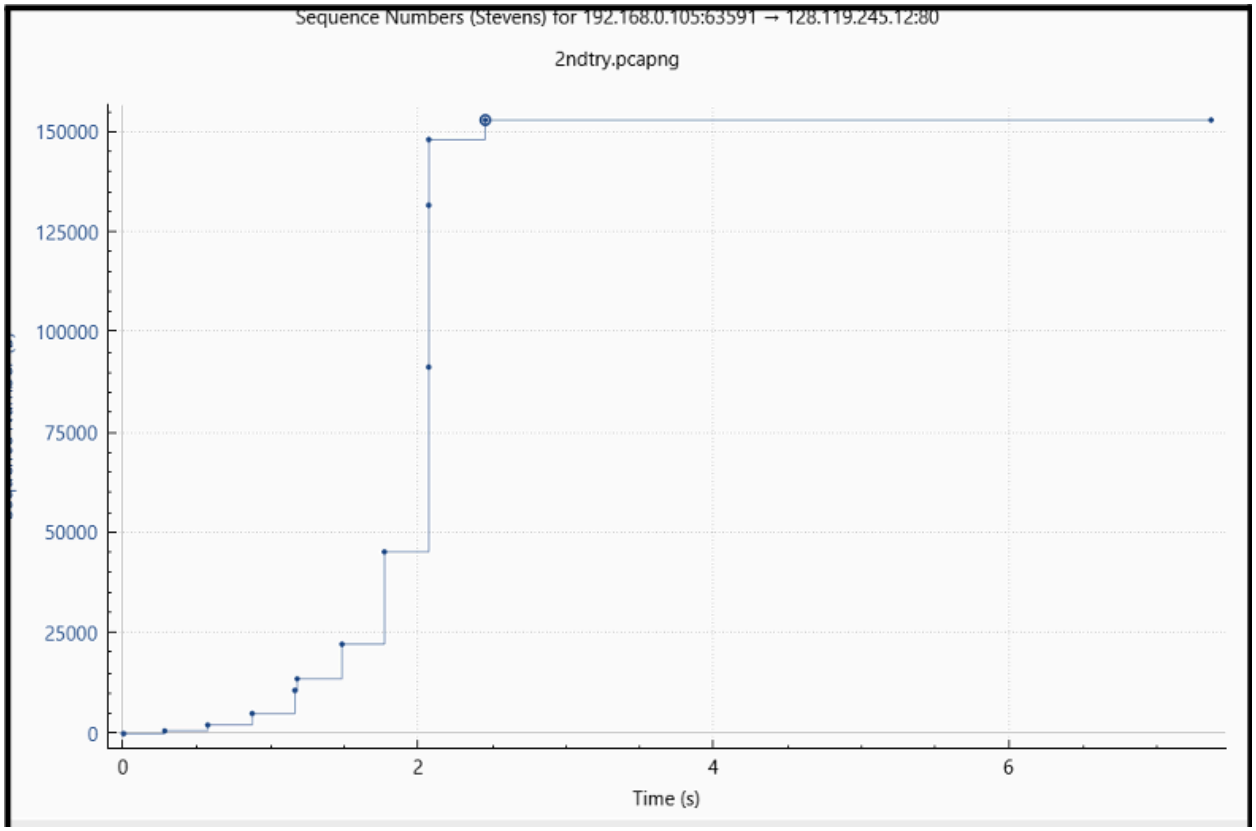
primarily slow-start during the fleets that pointed out; later the trace appears to move toward congestion-avoidance.

13. These “fleets” of segments appear to have some periodicity. What can you say about the period?
Ans:differences between the fleet times listed:
0.053 − 0.025 ≈ 0.028 s
- 0.082 − 0.053 ≈ 0.029 s
- 0.100 − 0.082 ≈ 0.018 s (one is a bit shorter, possibly measurement/granularity or slightly earlier retransmit/packing)

The two largest gaps give an average of about ~0.028–0.03 seconds (≈28–30 ms). This period is the round-trip time (RTT) for this connection — each fleet is spaced by roughly one RTT

14. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu
From the Time-Sequence-Graph (Stevens) plot, we can observe that the sequence number increases rapidly at the beginning, with each burst (“fleet”) of packets being larger than the previous one up to around t = 2 s. This pattern of exponential growth in the amount of data transmitted per RTT is characteristic of TCP’s Slow Start phase. After about t = 2 s, the curve flattens, indicating

that the sender has reached the congestion window limit or is transitioning to the Congestion Avoidance phase.



The fleets of packets appear to have a roughly periodic spacing — each fleet begins after approximately 0.25 to 0.3 seconds, which corresponds to the measured RTT in this trace. This periodicity reflects how each new group of packets is triggered by the acknowledgment (ACK) of the previous fleet. Thus, the fleets occur once per RTT, consistent with the expected TCP behavior during the Slow Start phase.