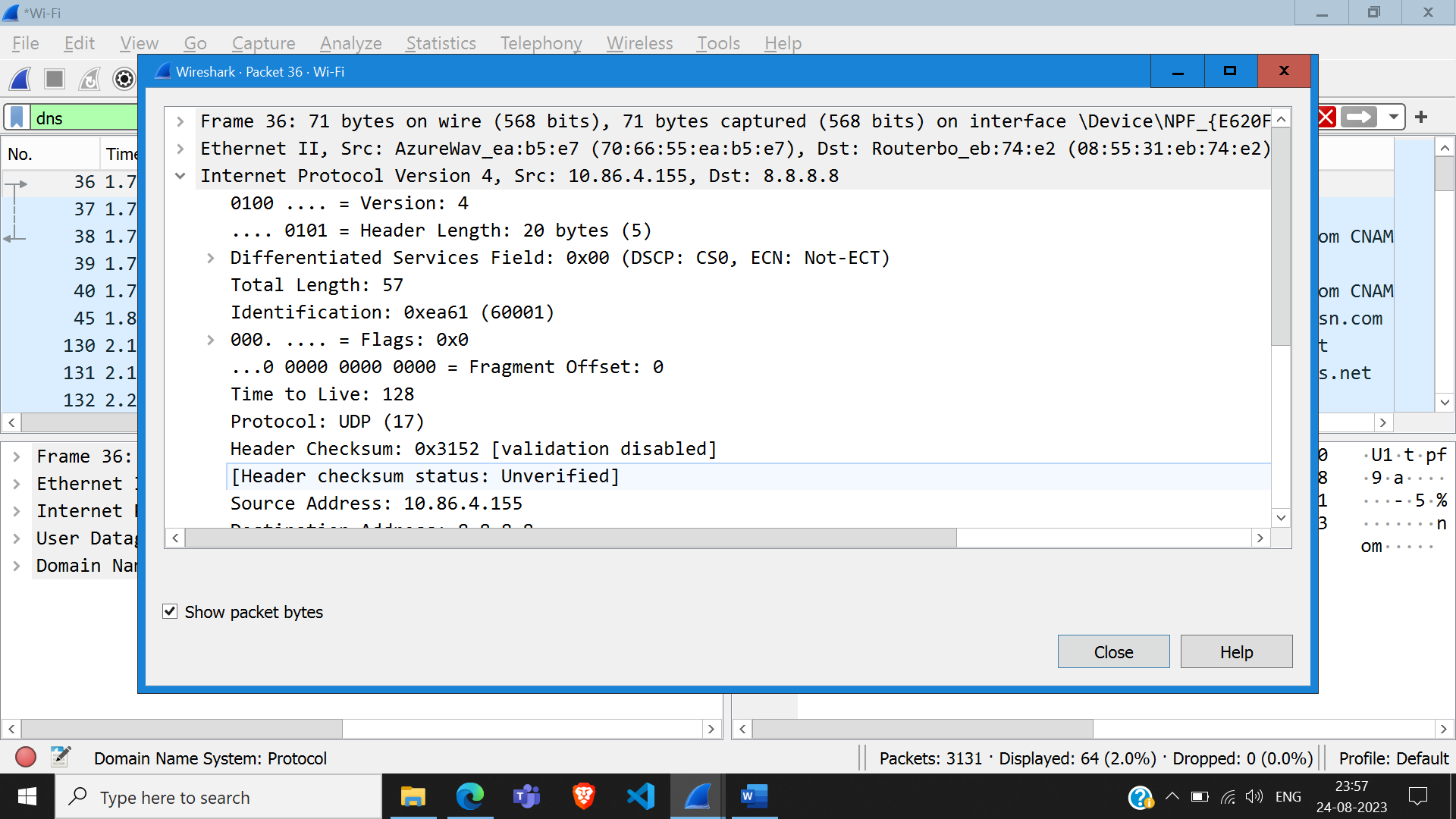
**Analyzing UDP datagrams using Wireshark:2A. In the packet list pane, select the first DNS packet. In the packet detail pane, select the User Datagram Protocol. The UDP hexdump will be highlighted in the packet byte lane. Using the hexdump, Answer the following:**

**A screenshot of a computer

Description automatically generated**



a. the source port number. Source Port: 51501

b. the destination port number. Destination Port: 53

c. the total length of the user datagram. Total Length: 57

d. the length of the data. Length: 37

e. whether the packet is directed from a client to a server or vice versa. Client to server

f. the application-layer protocol. UDP

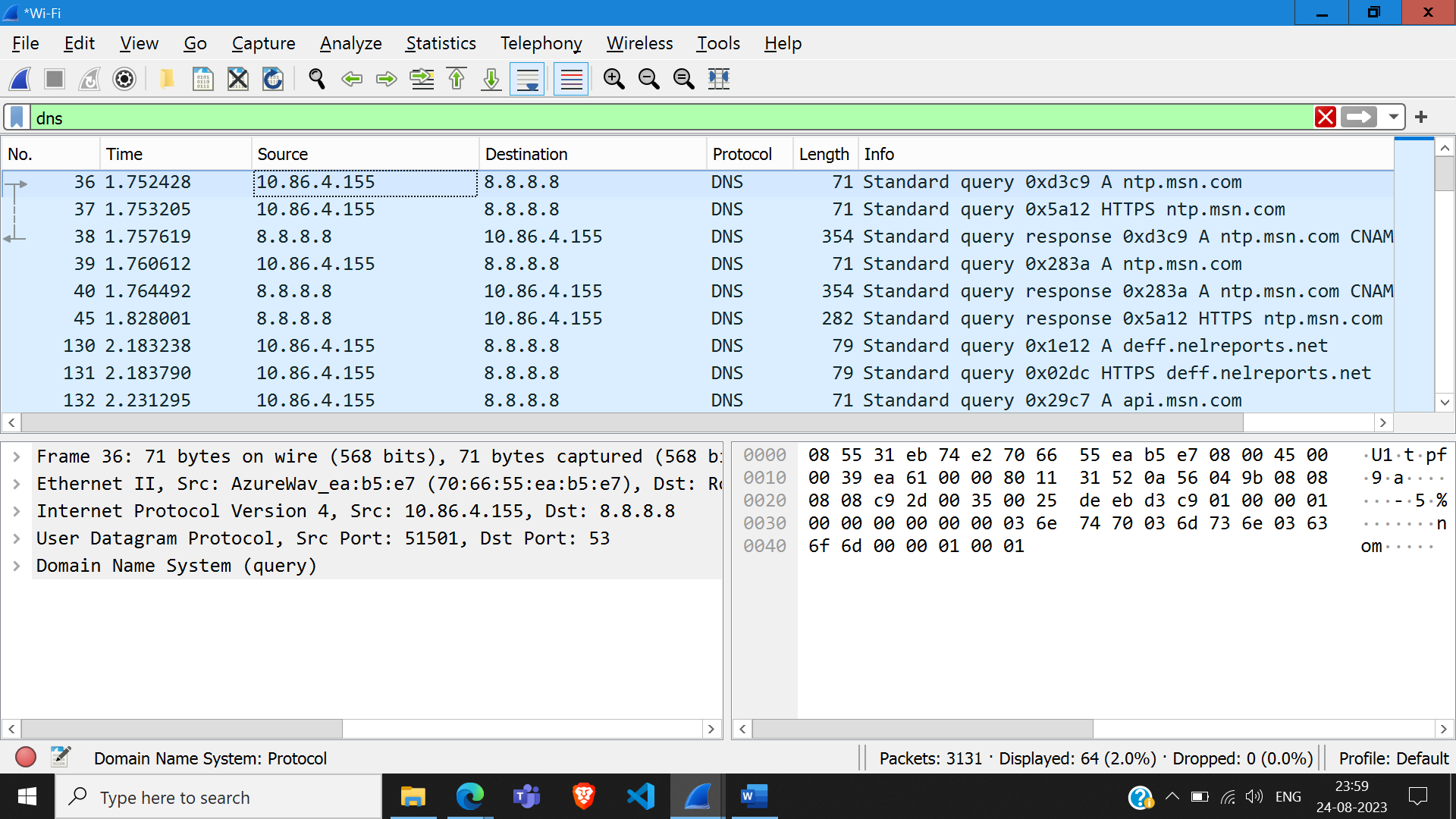
g. whether a checksum is calculated for this packet or not.

Checksum: 0xdeeb [unverified]

[Checksum Status: Unverified]

**2B. What are the source and destination IP addresses in the DNS query message? What are those**

**addresses in the response message? What is the relationship between the two?**



Source: 10.86.4.155 Destination:8.8.8.8

Destination: 10.86.4.155 Source:8.8.8.8

In the response, the addresses are exactly inverted (source is destination and destination is source)

**2C. What are the source and destination port numbers in the query message? What are those**

**addresses in the response message? What is the relationship between the two? Which port**

**number is a well-known port number?**

Source and destination port numbers in query: 51501, 53

Source and destination port numbers in response: 53, 51501

In the response, the port numbers are exactly inverted (source is destination and destination is source)

The standard port for DNS is 53.

**2D. What is the length of the first packet? How many bytes of payload are carried by the first**

**packet?**

Total Length = 57

Udp Payload = 29 bytes

**Analyzing TCP packets using Wireshark:**

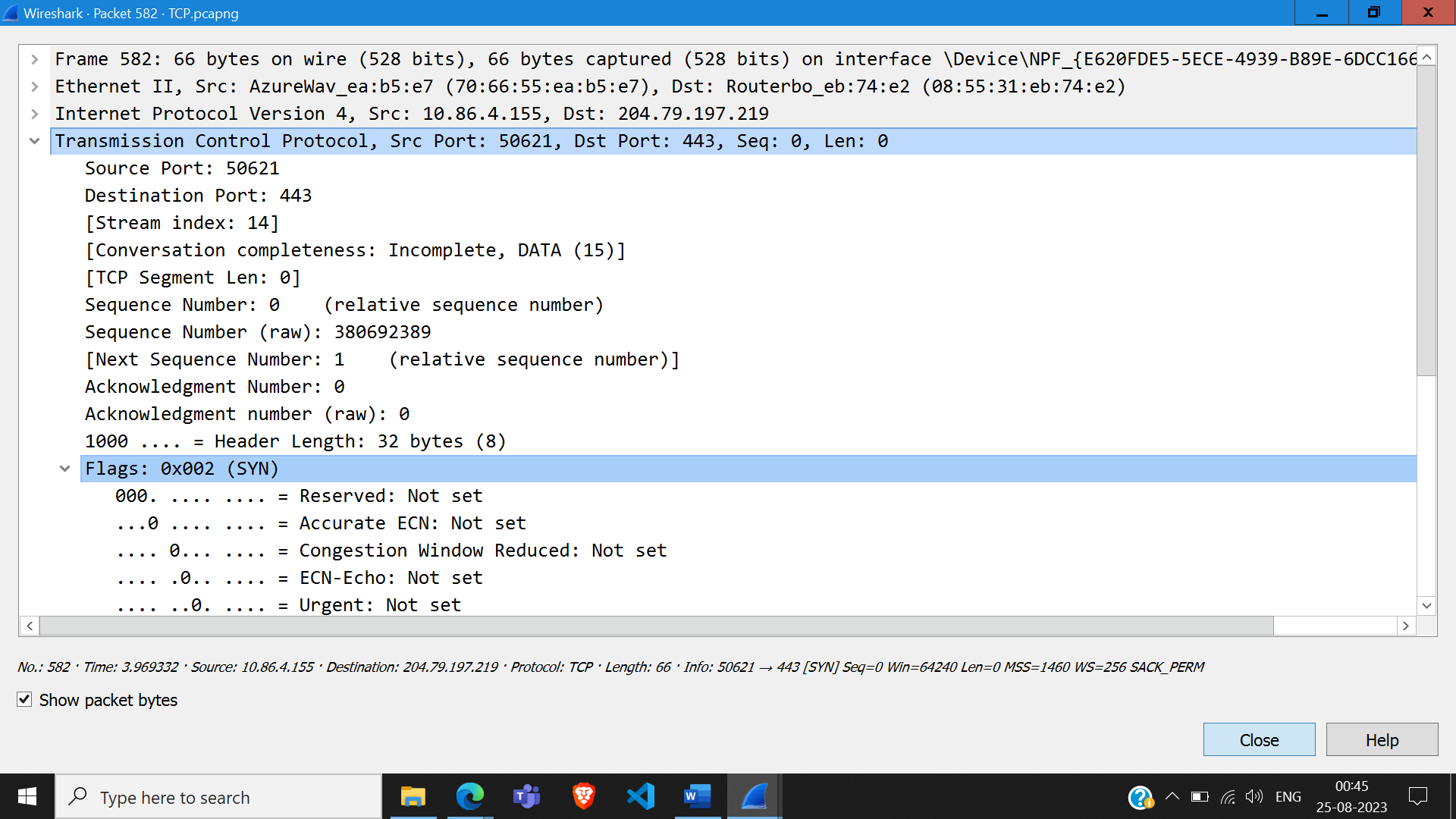
**Part I: Connection-Establishment Phase**

Identify the TCP packets used for connection establishment. Note that the last packet used for

connection establish may have the application-layer as the source protocol.

Questions : Using the captured information, answer the following question in your lab report about packets used for connection establishment

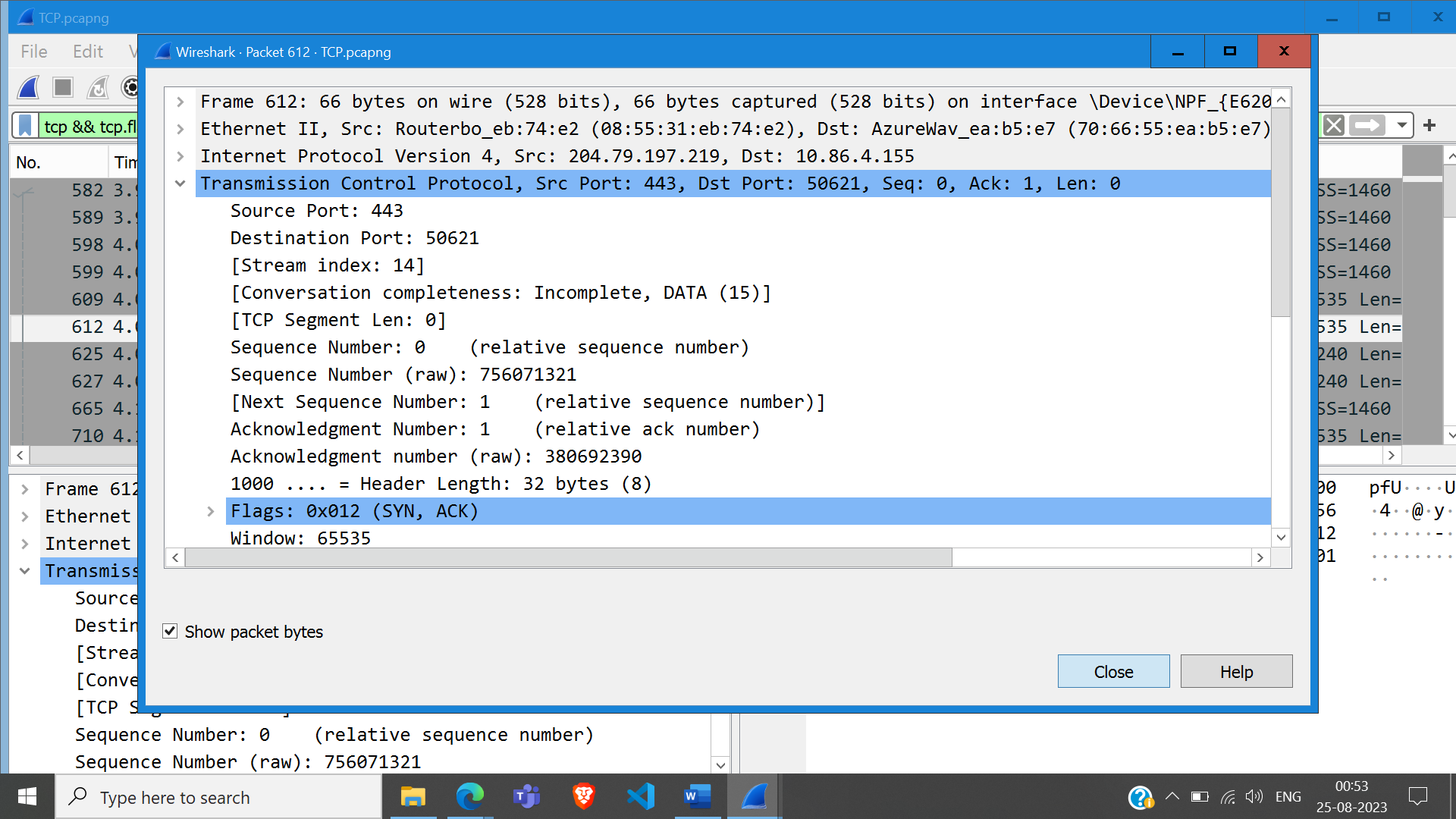
**SYN :**



A computer screen shot of a computer

Description automatically generated

**SYN-ACK:**



A screenshot of a computer

Description automatically generated

1. What are the socket addresses for each packet?

• SYN: Source: 10.86.4.155, Destination: 204.79.197.219

• SYN-ACK: Source: 204.79.197.219, Destination: 10.86.4.155

2. What flags are set in each packet?

• SYN: 0x002 (SYN) (SYN flag is set)

• SYN-ACK: 0x012 (SYN-ACK) (SYN and ACK flags are set)

3. What are the sequence number and acknowledgment number of each packet?

• SYN: Sequence Number : 380692389 , Ack Number: 0

• SYN-ACK: Sequence Number: 756071321 , Acknowledgment number : 380692390

4. What are the window size of each packet?

• SYN: 64240

• SYN-ACK: 65535

**Part II: Data-Transfer Phase**

1. What TCP flags are set in the first data-transfer packet (HTTP GET message)?

ACK and PUSH flags are set

2. How many bytes are transmitted in this packet?

154 bytes

3. How often does the receiver generate an acknowledgment?

0.04 seconds

4. How many bytes are transmitted in each packet? How are the sequence and acknowledgment numbers related to number of bytes transmitted?

154 bytes are transmited in the GET request, and 187 bytes are received in the OK response. The sequence number increases by the number of bytes transmited, and the acknowledgment number increases by the number of bytes received.

5. What are the original window sizes that are set by the client and the server? Are these numbers expected? How do they change as more segments are received by the client?

The window size when sending request to the server was set at 65536. The window size when receiving response was 98304. The window size will increase will a certain capacity (limited by hardware) to prevent congestion.

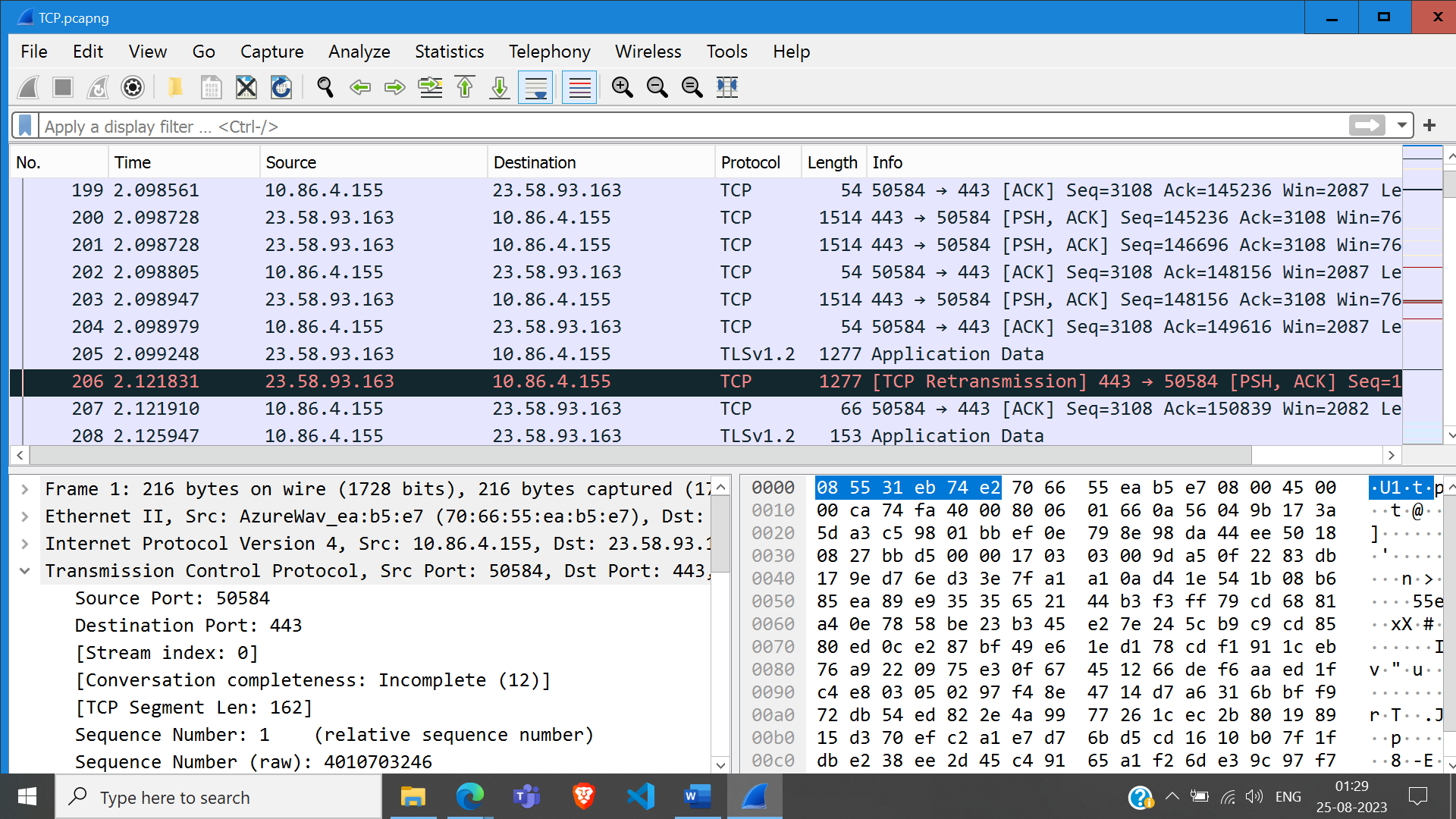
6. Explain how the window size is used in flow control?

In a sliding window system in TCP, the size of the window is governed by 2 things: o The size of the send buffer on the sending system o The size and available space in the receive buffer on the receiving system To avoid congestion, the sender cannot send more bytes than the space available in the receive buffer of the receiver. The sender must wait till the bytes in the receiving buffer have been acknowledged. This prevents congestion, and helps in flow control.

7. What is the purpose of the HTTP OK message in the data transfer phase?

The HTTP OK message is a feedback about the request that was previously sent. An OK response means that the request has succeeded.

**Part III: Connection Termination Phase**



1. How many TCP segments are exchanged for this phase?

4 segments are exchanged in the connection termination phase. (PSH, ACK, PSH, ACK)

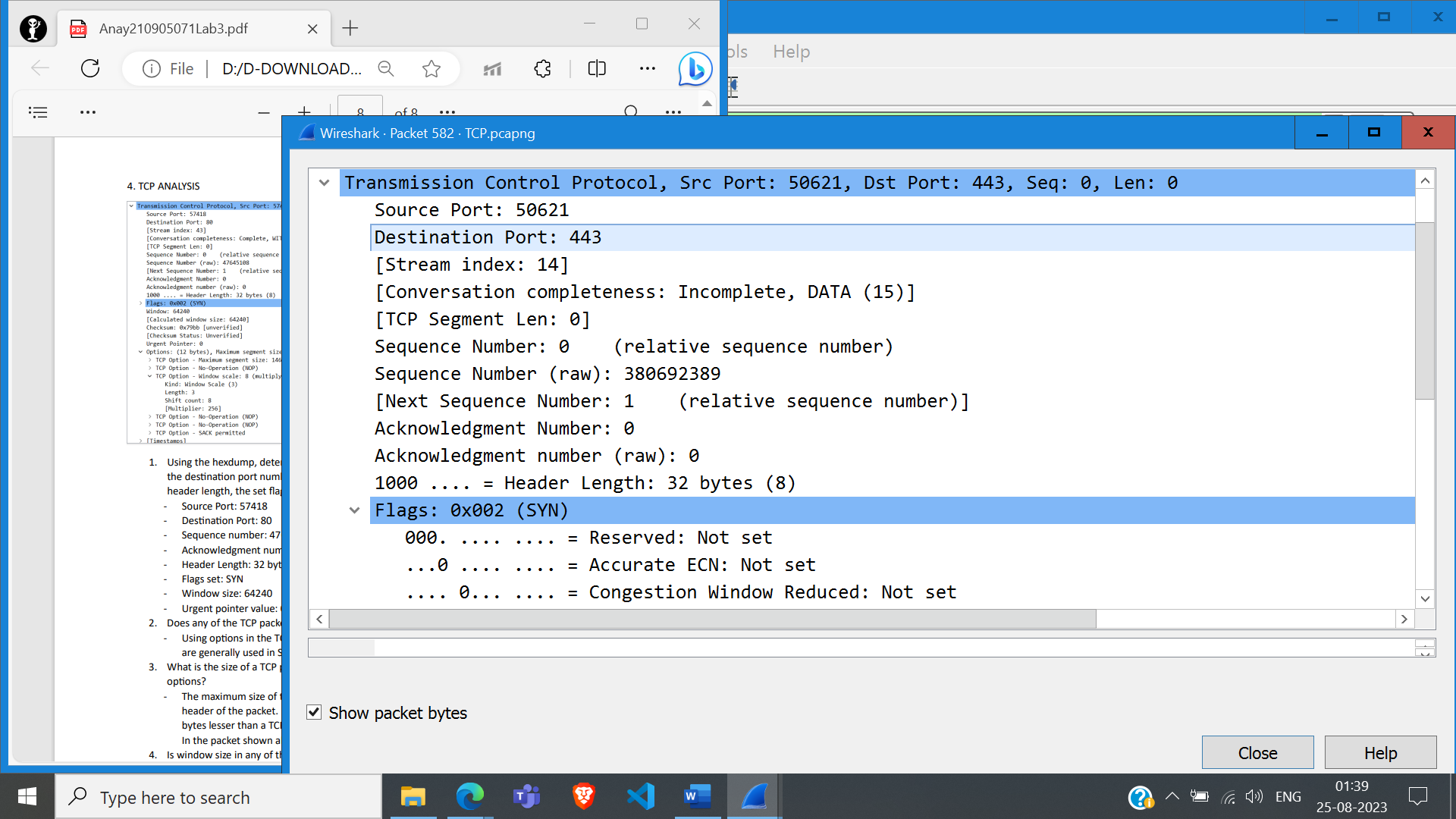
2. Which end point started the connection termination phase?

The server started the connection termination phase

3. What flags are set in each of segments used for connection termination?

First the server sends a segment with PSH, and ACK flags set. Then the client responds with a segment with PSH, and ACK flags set. Then the client sends a segment with ACK flag set, and finally the server sends a segment with PSH, and ACK flags set.

**II. LAB EXERCISES**



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. From the captured information, answer the following question in your lab report.

i. Using the hexdump, determine the following for any TCP packet: The source port number, the destination port number, the sequence number, the acknowledgment number, the header length, the set flags, the window size and the urgent pointer value.

- Source Port: 50621

-Destination Port: 443

- Sequence number: 380692389

- Acknowledgment number: 0

- Header Length: 32 bytes

- Flags set: SYN

- Window size: 64240

- Urgent pointer value: 0

ii. Using the information in the detail pane lane, verify your answers is question 1.

Verified

iii. Does any of the TCP packet header carry options? Explain your answer.

Using options in the TCP packet header is optional. Options are used for optimization and are generally used in SYN packets.

iv. What is the size of a TCP packet with no options. What is the size of a TCP packet with options?

The maximum size of the options field is 40 bytes. This is added onto the size of the header of the packet. Hence, at max, the size of a TCP packet with no options will be 40 bytes lesser than a TCP packet with options. In the packet shown above, the size of options is 12 bytes.

v. Is window size in any of the TCP packet zero? Explain your answer.

The window size may be zero if the receiver buffer is full, and receiver tells the sender to keep the connection open, but not to send it anything, as it is already overwhelmed. Once the receiver can handle more data, it can advertise a window size greater than zero. Another reason for window size being zero in a TCP packet could be because of the RST flag. Sometimes connections are forcefully terminated with a packet with RST flag, and these packets have a window size of zero.

2. Analyze Interaction between your TCP Client-Server Programs using wireshark

Done

3. Analyze Interaction between your UDP Client-Server Programs using wireshark

Done