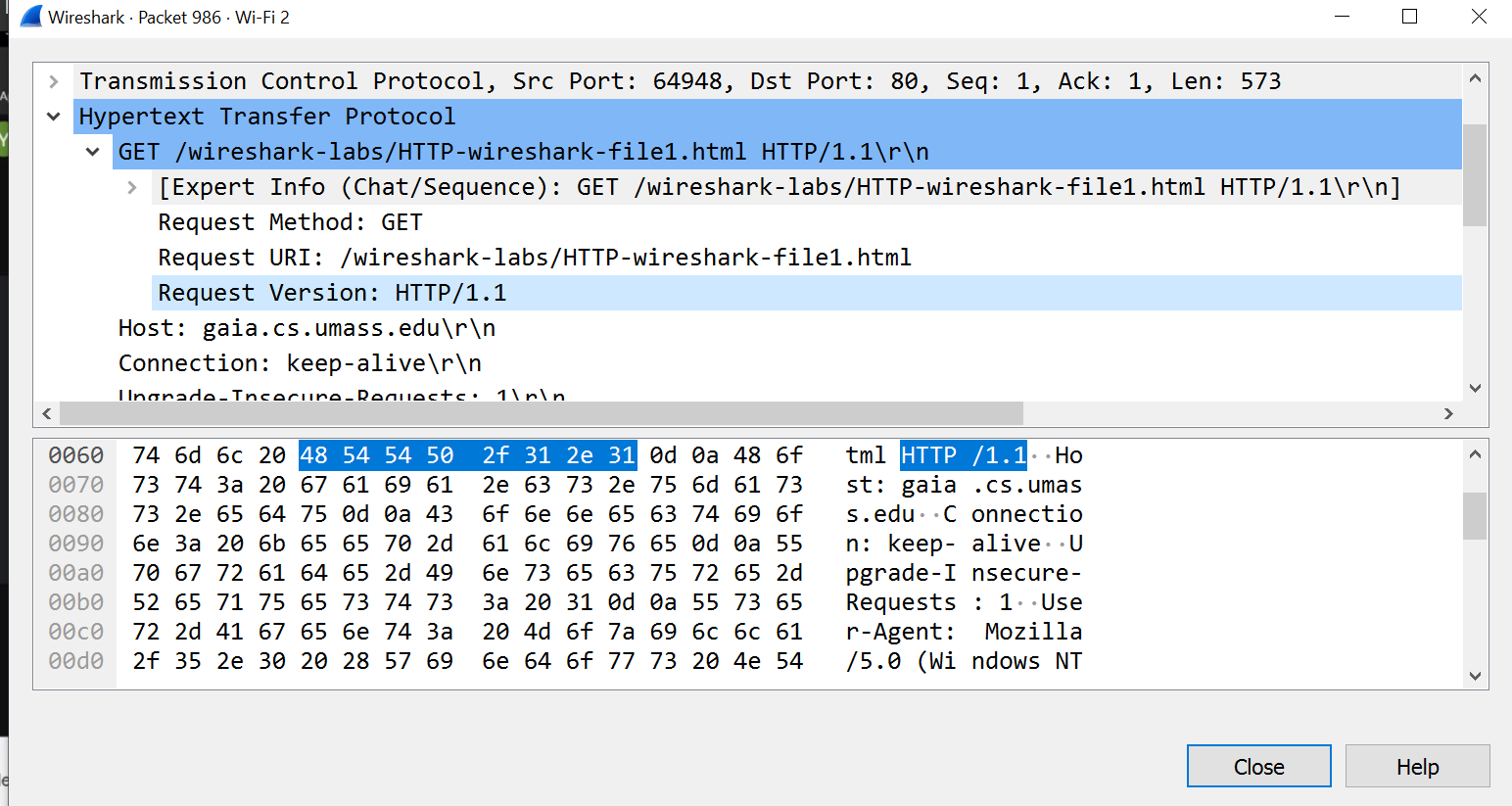
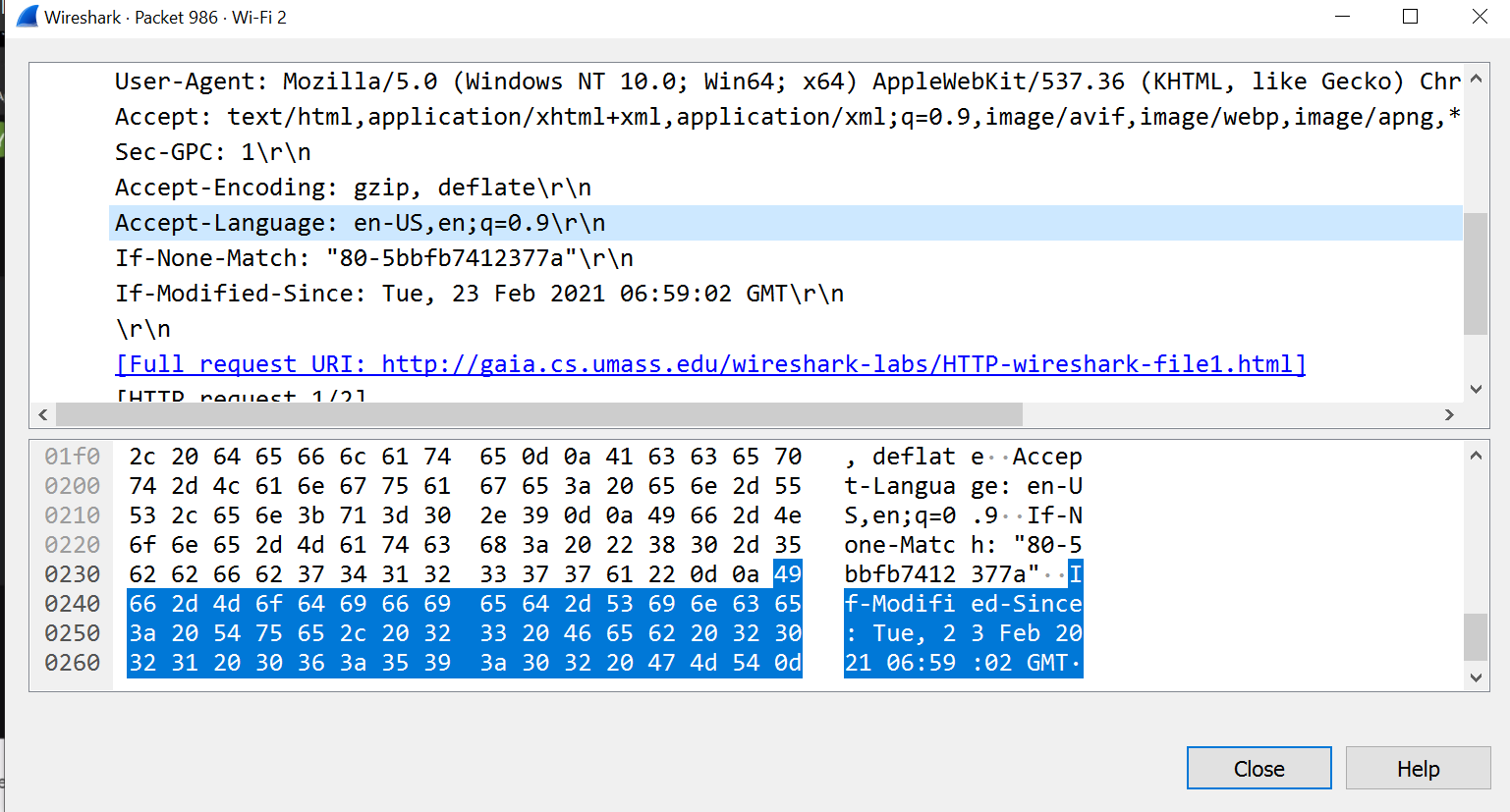
1. Is your browser running HTTP version 1.0 or 1.1?

**Ans:** Its running on HTTP version 1.1



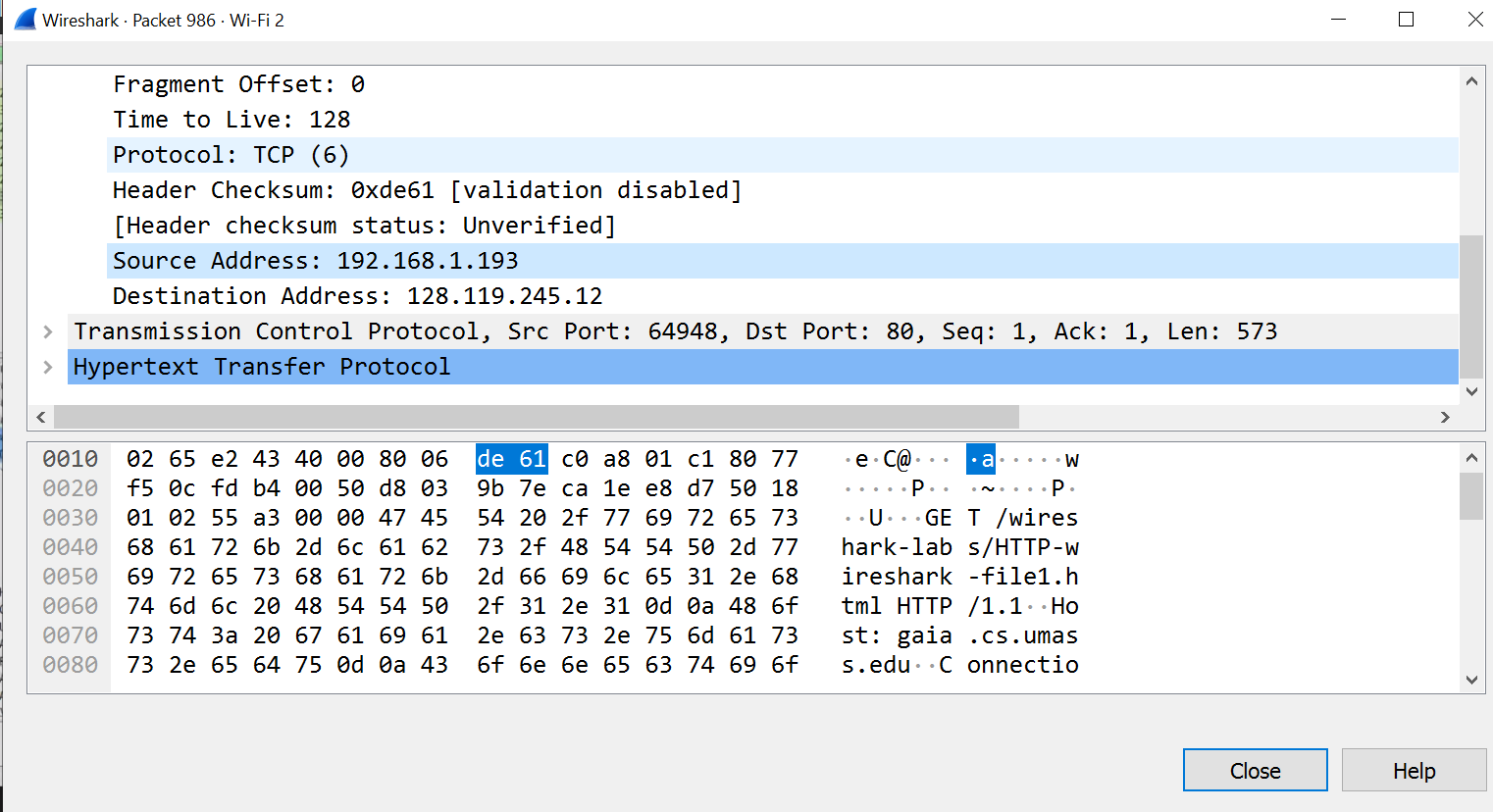
2. What languages (if any) does your browser indicate that it can accept to the server?

**Ans:** Its accepting English-US and English.



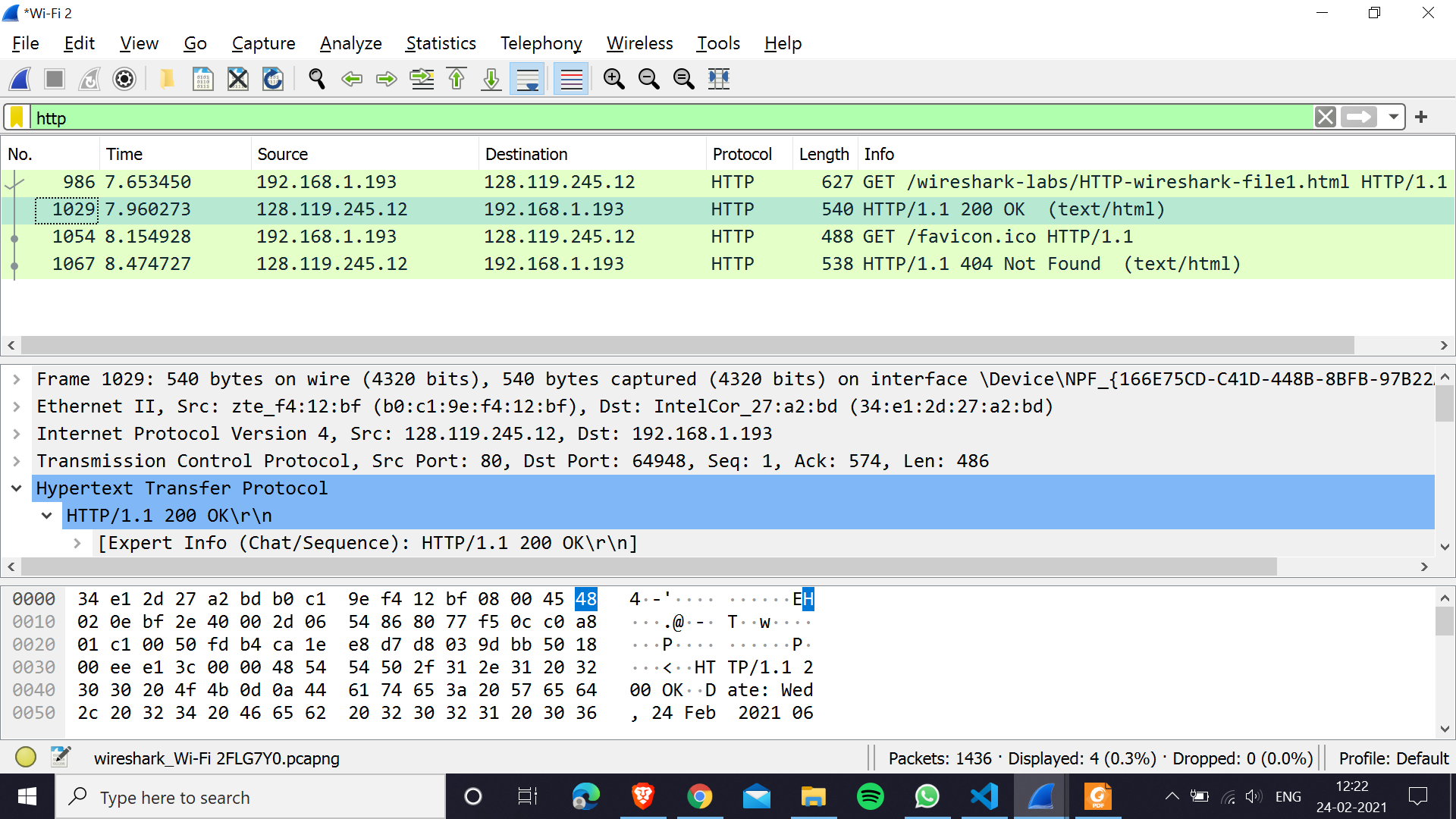
3. What is the IP address of your computer?

**Ans:** IP address of my computer is 192.168.1.193



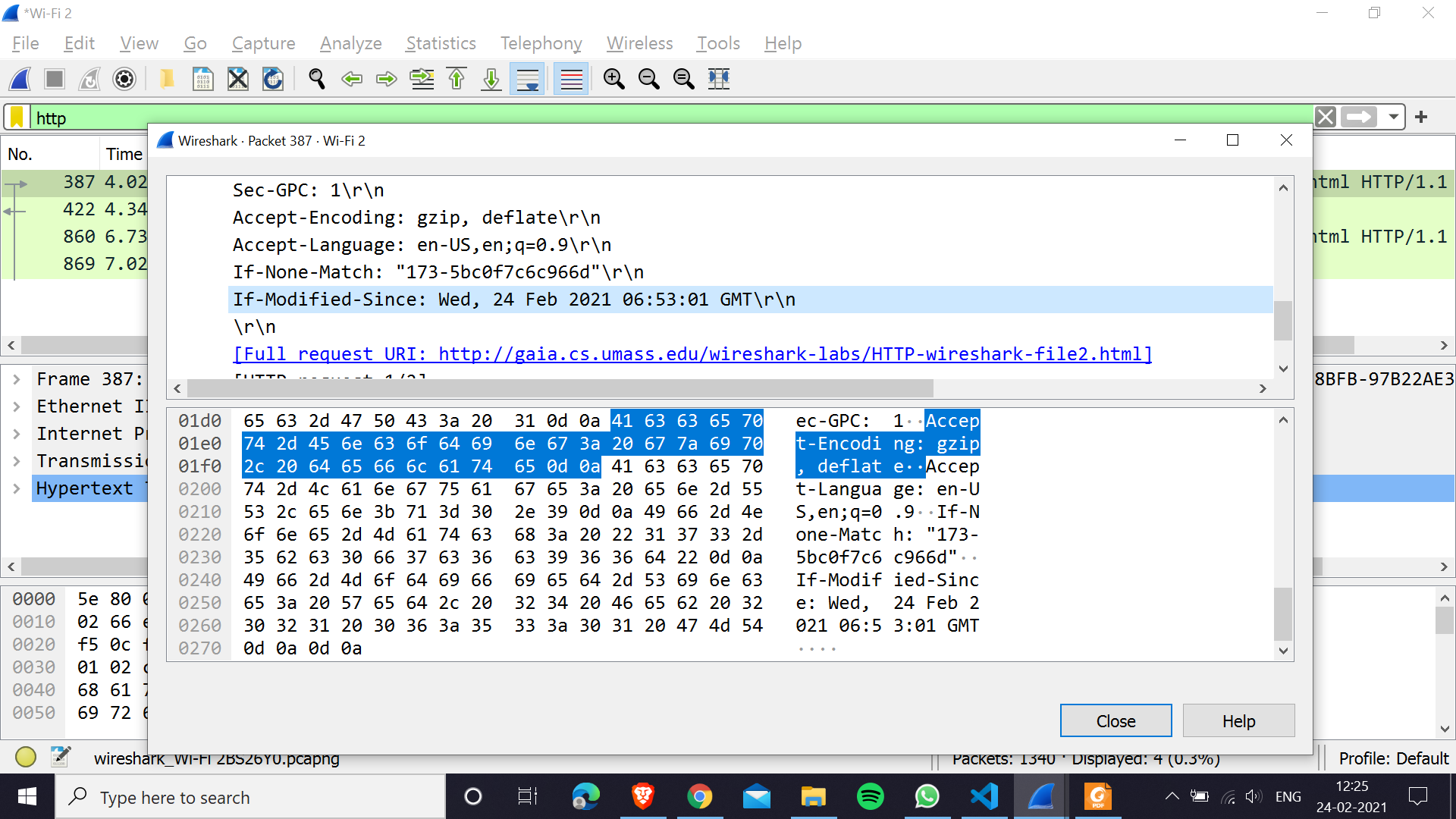
4. What is the status code returned from the server to your browser?

Ans: Its 200 OK status code.



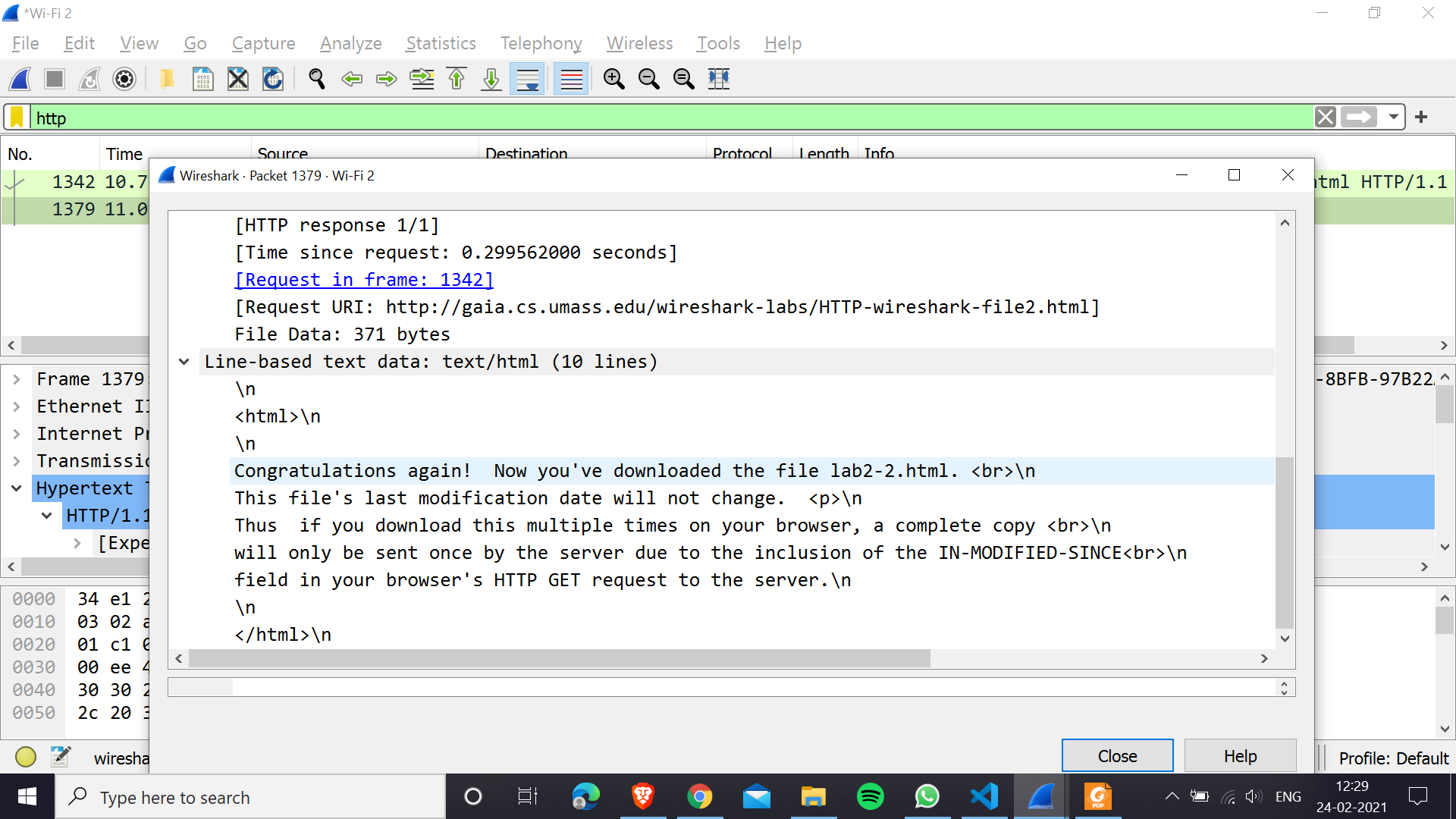
5. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET?

Ans: Yes, it holds the time after which, if the requested source was modified, it would be sent by the server, otherwise it won’t.



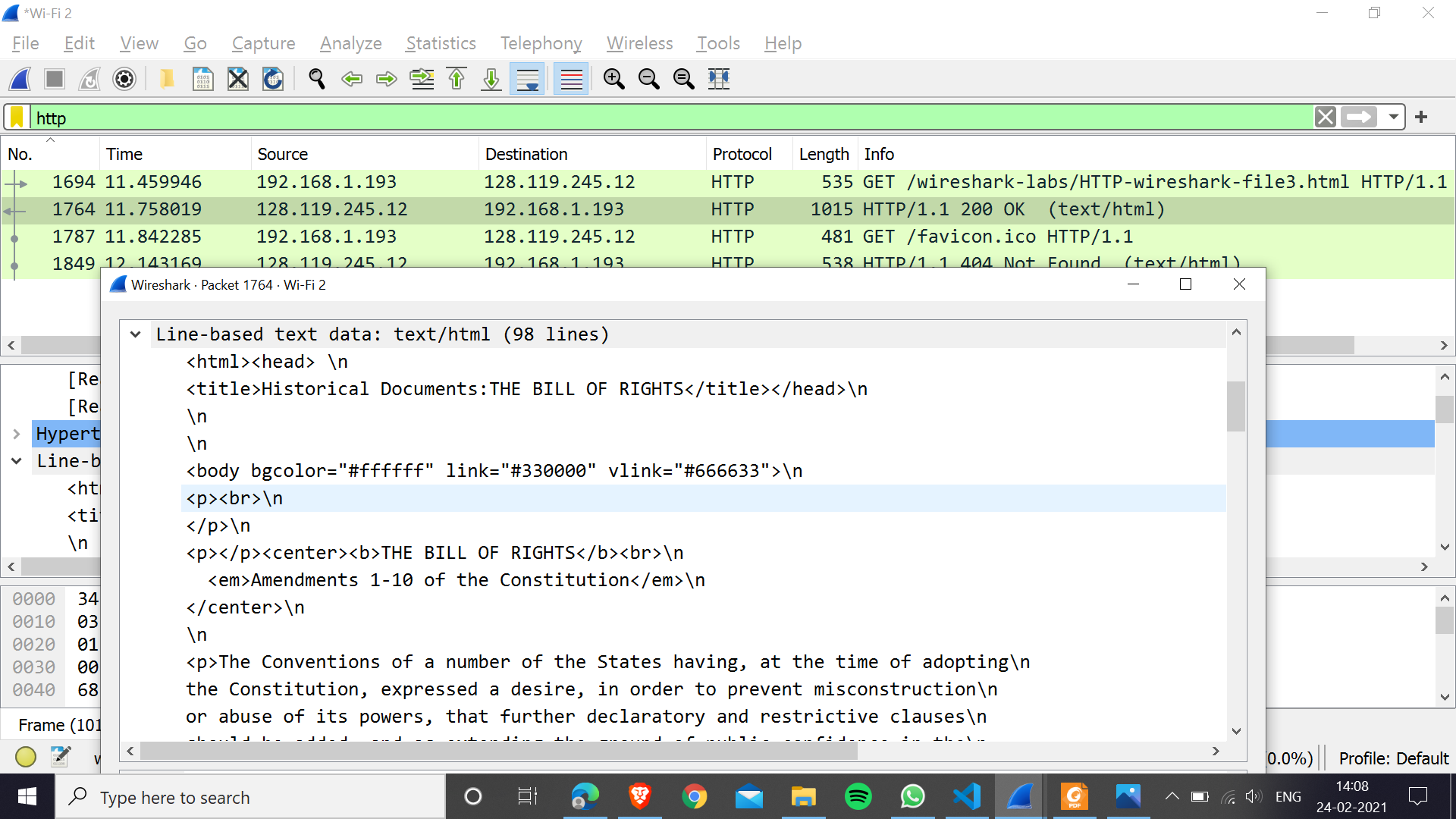
6. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?

Ans: Yes, it did. It was sent as a text/html file inside the HTTP data



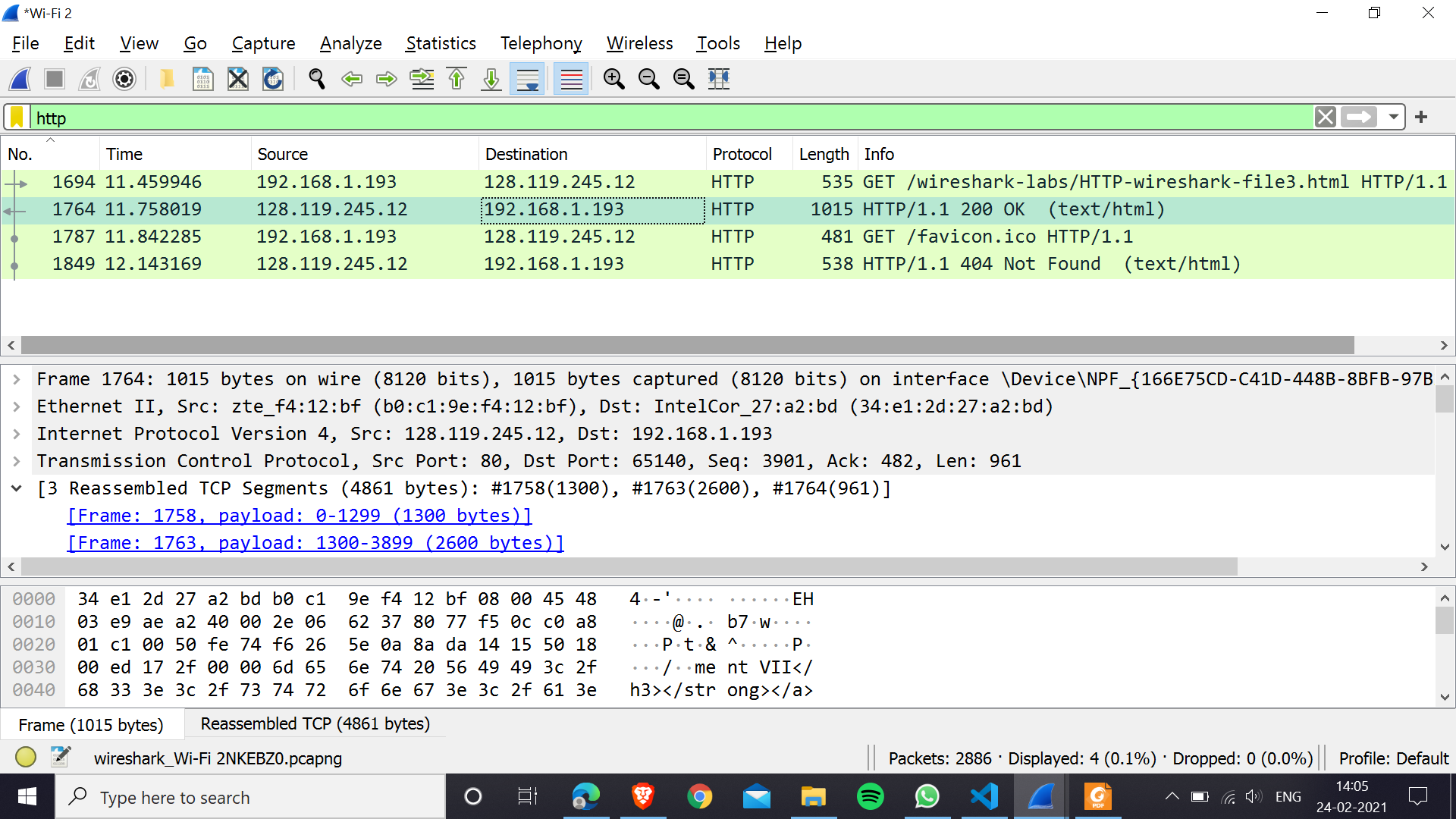
7. How many HTTP GET request messages did your browser send? Which packet number in the trace contains the GET message for the Bill or Rights?

Ans: There were 2 GET messages (one for file3.html and one for favicon.ico). The packet number 1694 holds the GET message for the Bill of Rights.



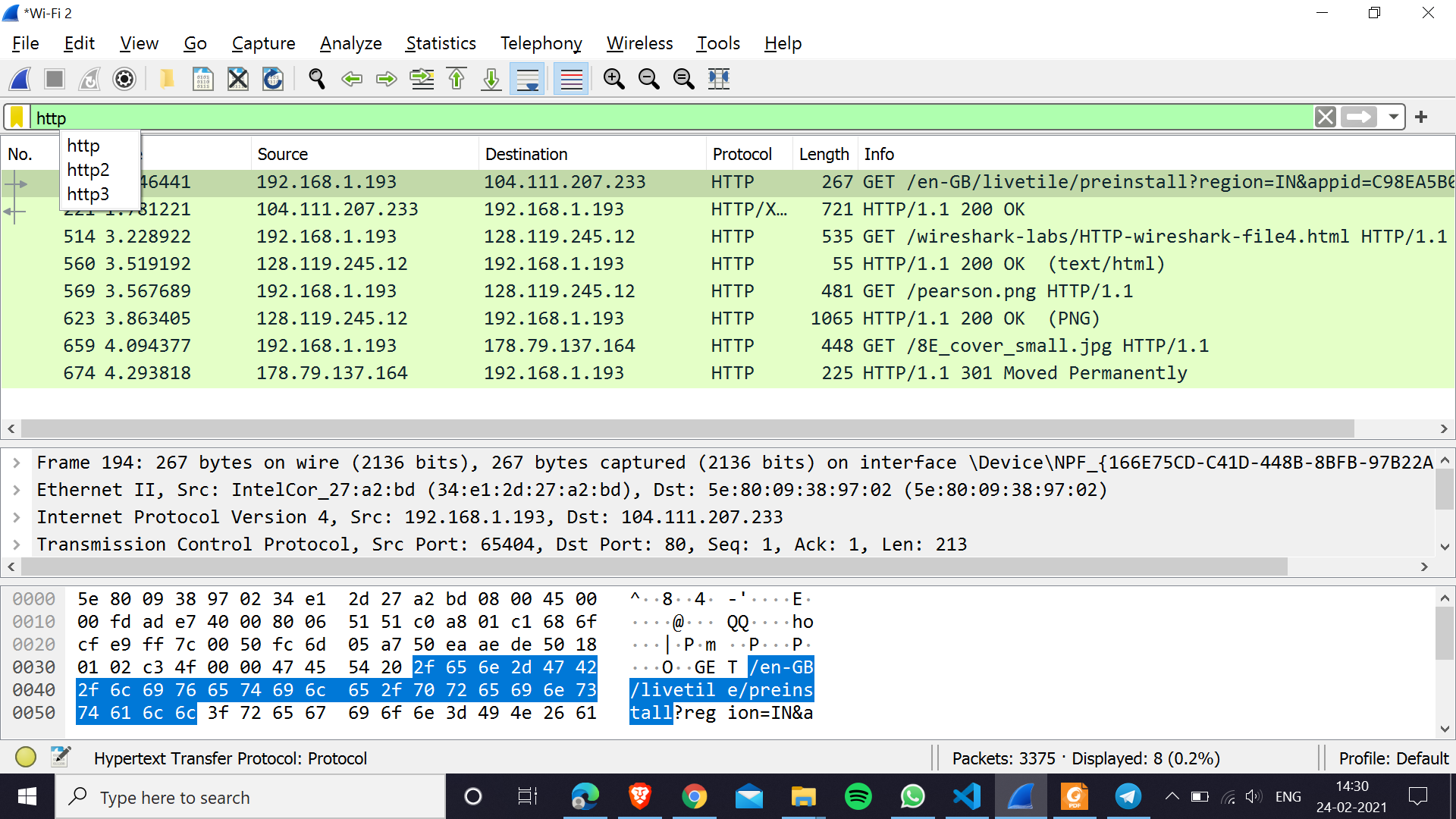
8. Which packet number in the trace contains the status code and phrase associated with the response to the HTTP GET request?

Ans: The packet number 1764 contains the status code 200 OK along with the response, i.e., the text/html file.



9. How many HTTP GET request messages did your browser send? To which Internet addresses were these GET requests sent?

Ans: 4 HTTP GET request messages were sent to the IP addresses 104.111.207.223, 128.119.245.12 (twice), 178.79.137.164

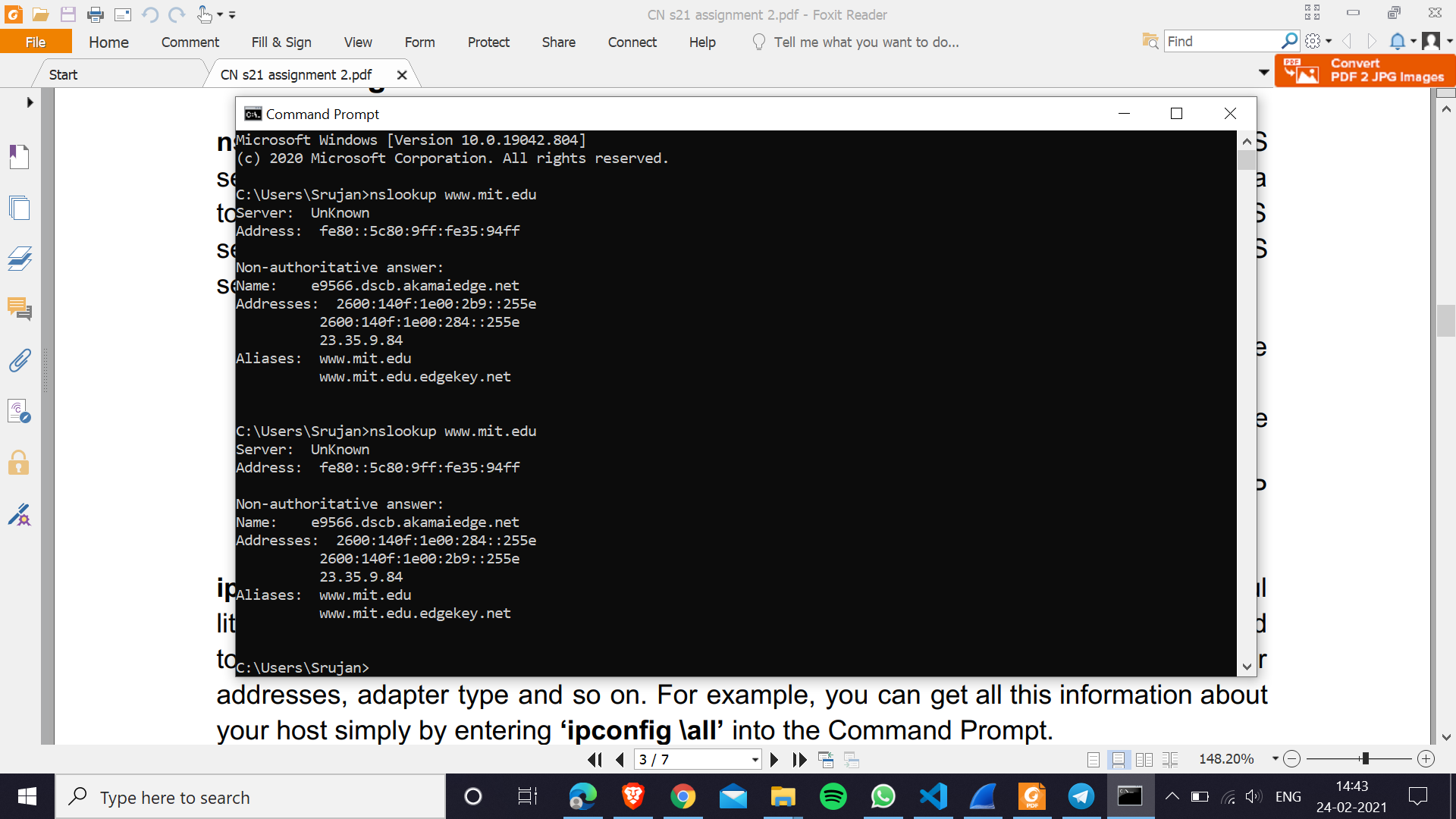


10.Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two websites in parallel? Explain

Ans: They were downloaded serially. We can see the timestamps of the messages in the above screenshot that after the response of the first image was received, the request for the second image was sent. This is because the network pipe at a time contains packets from only 1 type of message (a response, a request, an ACK).

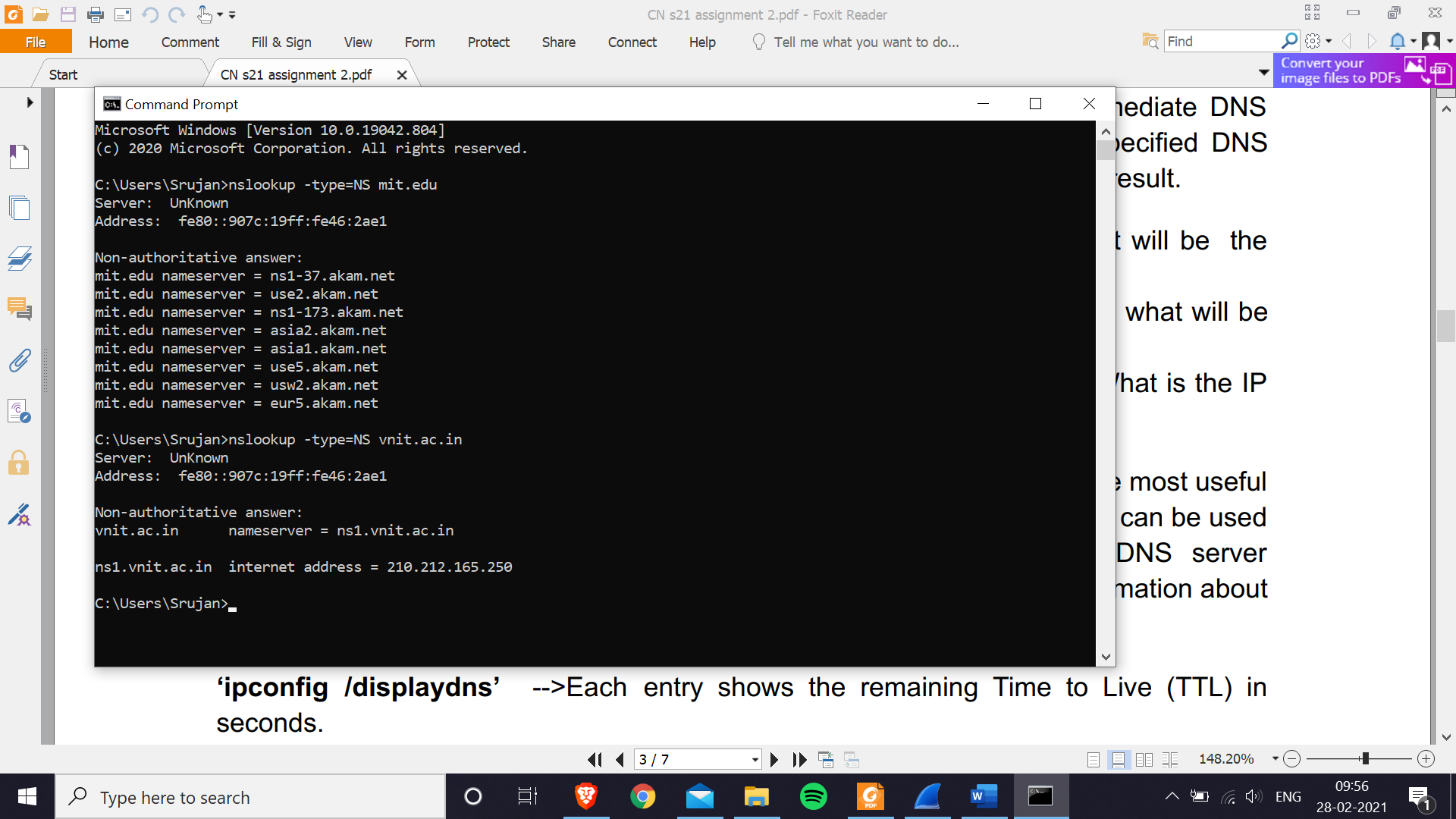
1. Run ‘nslookup www.mit.edu’ on your command prompt and what will be the name and IP address of the DNS server that provides the answer?

**Ans:** Name: Unknown, Address: fe80::5c80:9ff:fe35:94ff (IPv6 format)



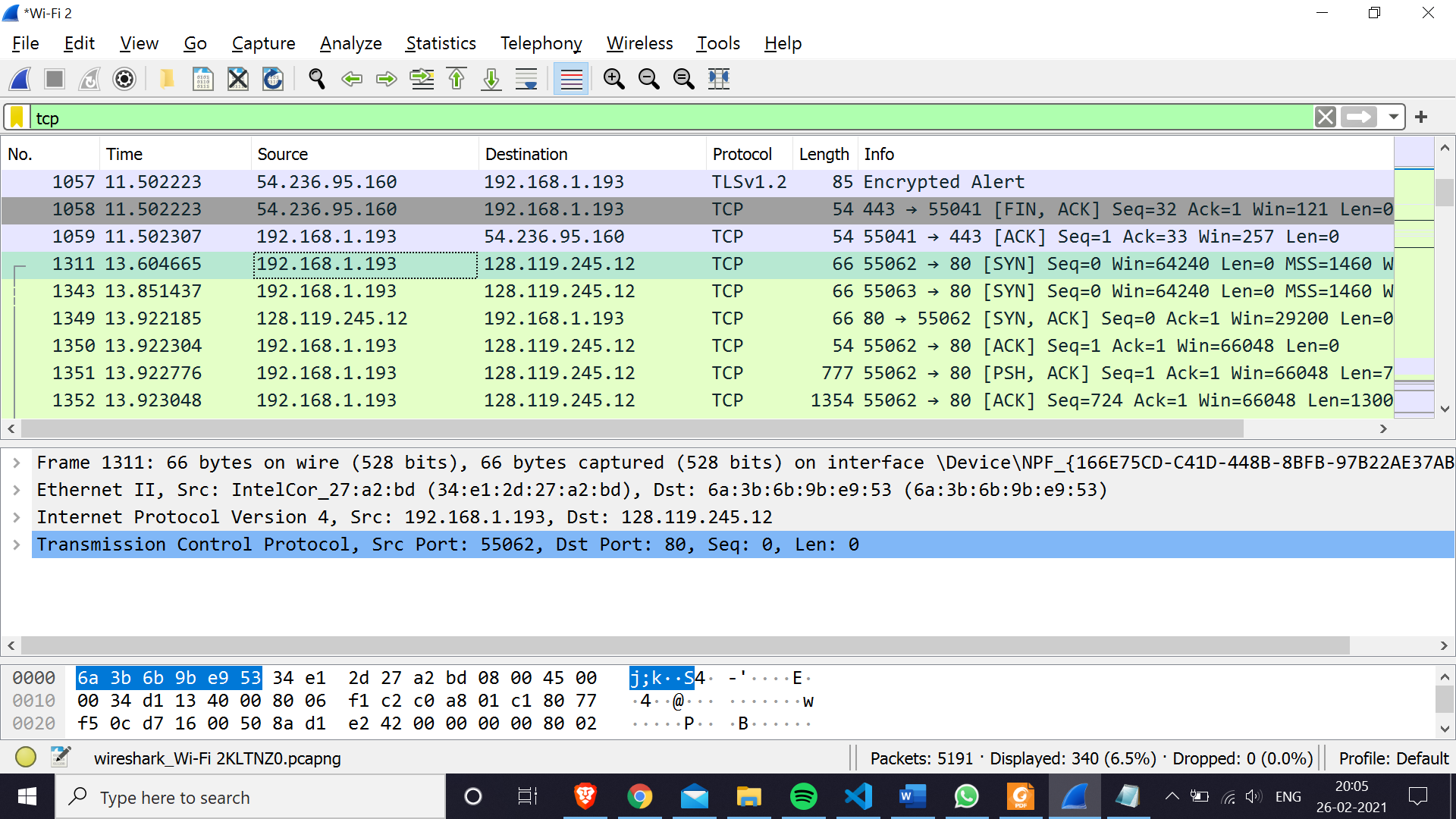
2. Run ‘nslookup –type=NS mit.edu’ on your command prompt and what will be the host names of the authoritative DNS for mit.edu.

**Ans:** The command shows only “Non-authoritative” meaning that these server-names came from a non-authoritative server’s cache rather than from an authoritative MIT DNS server.



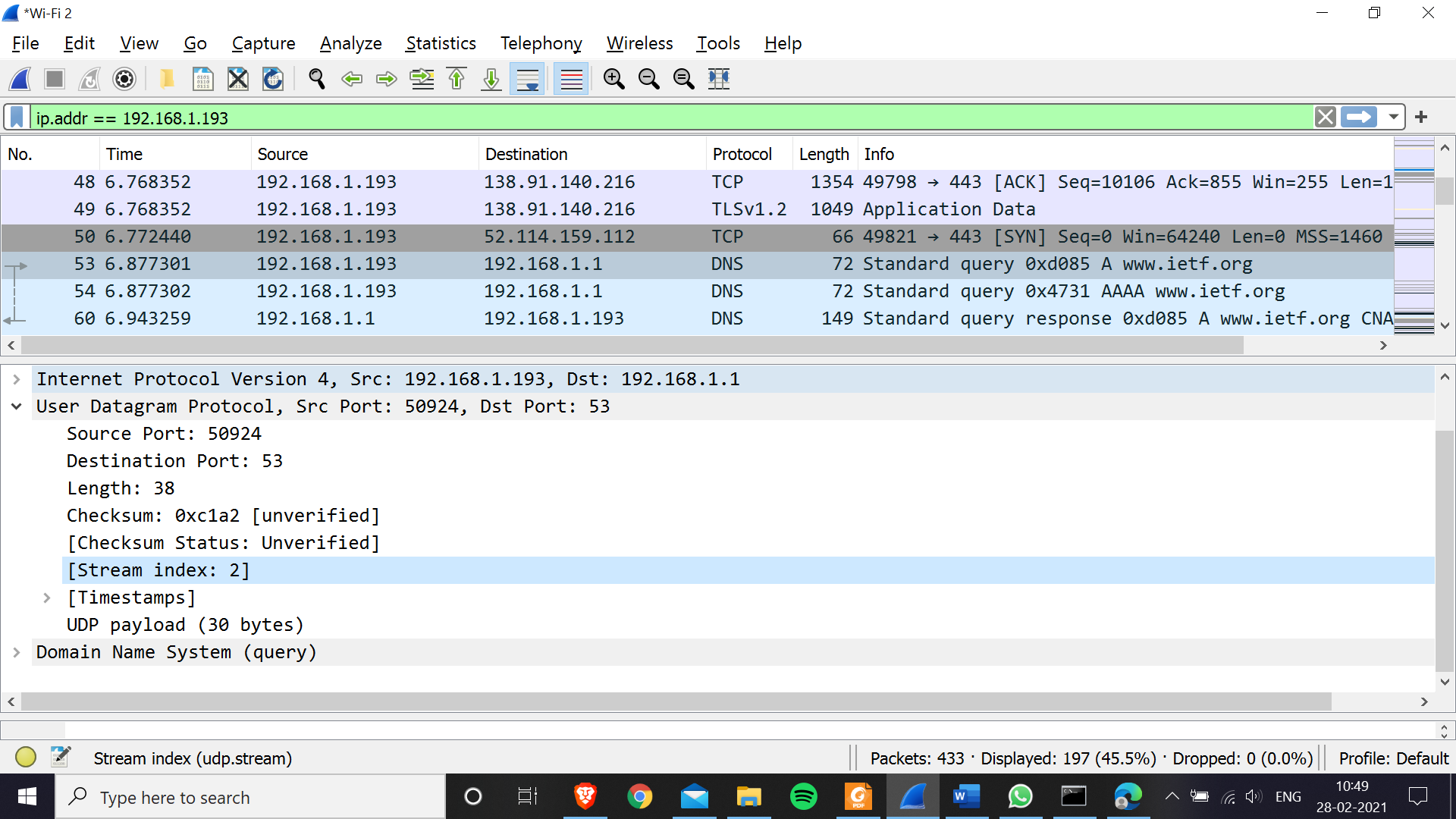
3. Run nslookup to obtain the IP address of a Web server in Asia. What is the IP address of that server?

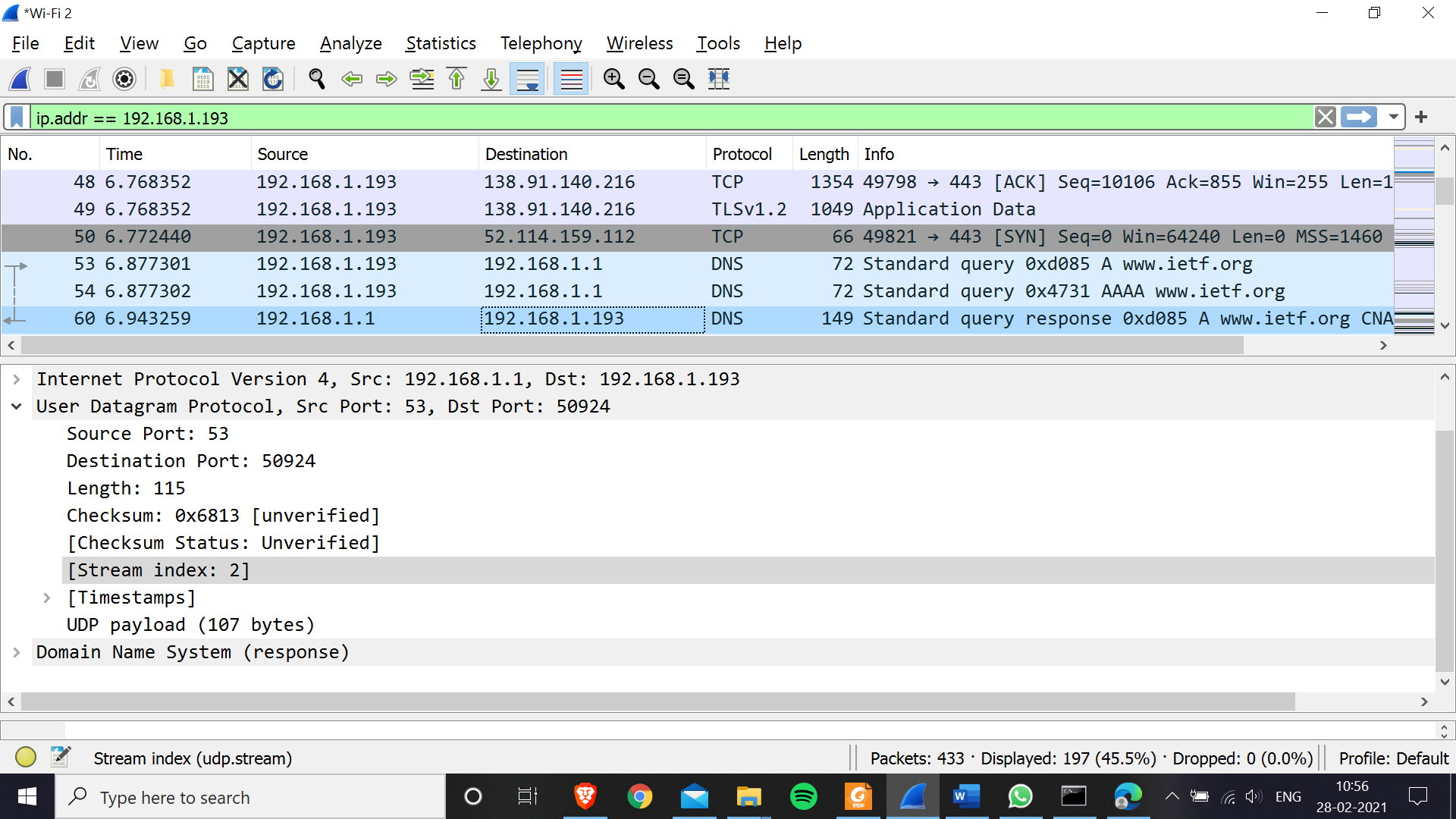
**Ans:** I did this for our college website vnit.ac.in. IP address of the website is 210.212.165.250 (as seen in the above screenshot).



4. Locate the DNS query and response messages. Are they sent over UDP or TCP?

Ans:The DNS query and response messages are sent over UDP (packet numbers 53,54,56 in the below screenshot).



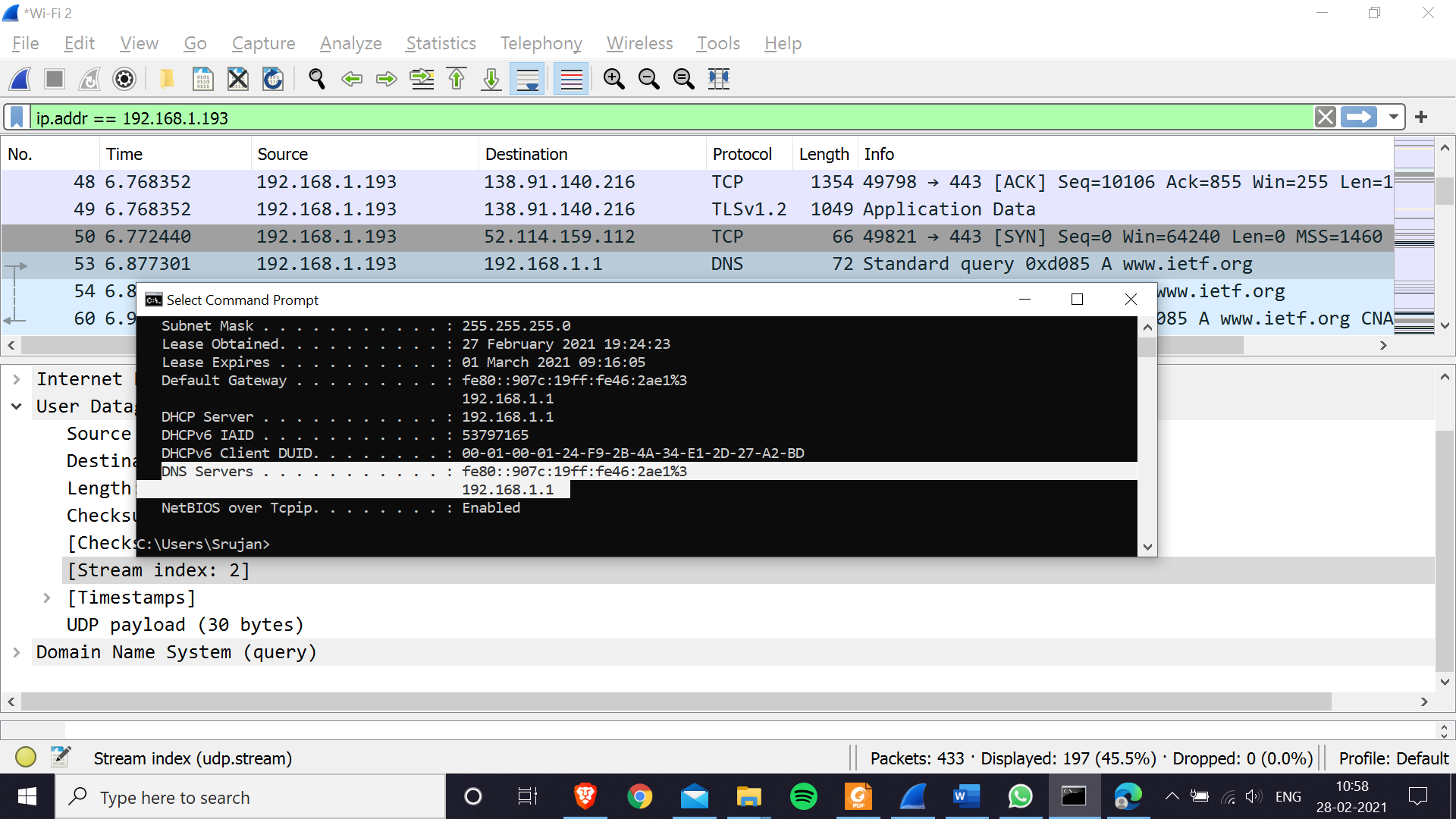


5. What is the destination port for the DNS query message? What is the source port of the DNS response message?

Ans:destination port is 53 in DNS query and source port is 53 in DNS response image (as seen in above screenshots).

6. To what IP address is the DNS query message sent? Use ipconfig to determine the IP address of your local DNS server. Are these two IP addresses the same?

Ans: The DNS query was sent to IP address 192.168.1.1. Yes, it is the same IP address as that of my local DNS server.



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

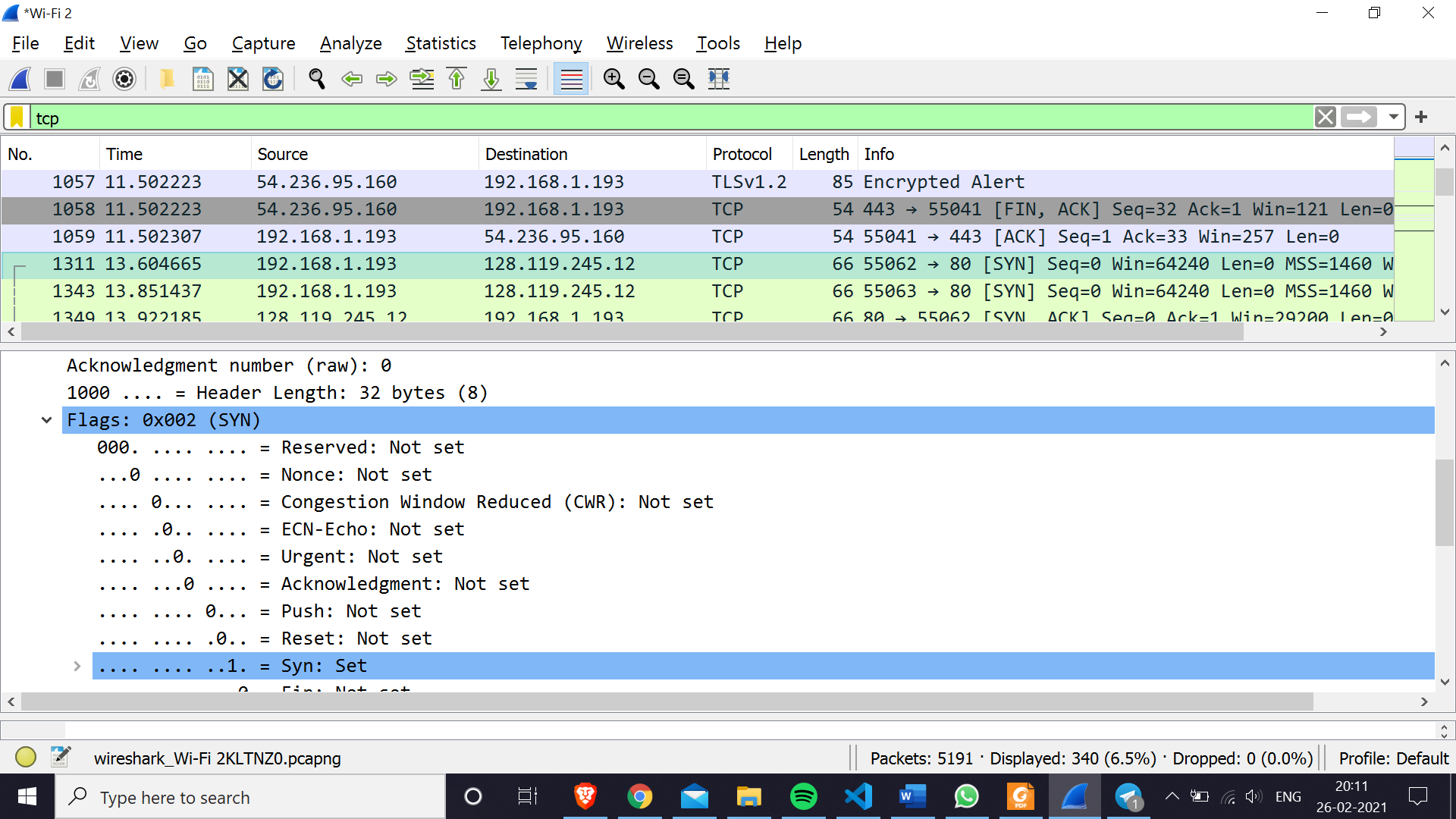
**Ans:** IP address is 192.168.1.193 and the source port is 55062 (as seen in above screenshot).

**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 of gaia.cs.umass.edu is 128.119.245.12 and the port number its using is 80 (as seen in above screenshot).

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

**Ans:** The sequence number is zero (Seq=0) and the flags are set such that SYN is set (as seen in below screenshot).



**4.** Answer the following:  
**a.** What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN?

**Ans:** Seq = 0 for the SYNACK segment.

**b.** What is the value of the Acknowledgement field in the SYNACK segment?

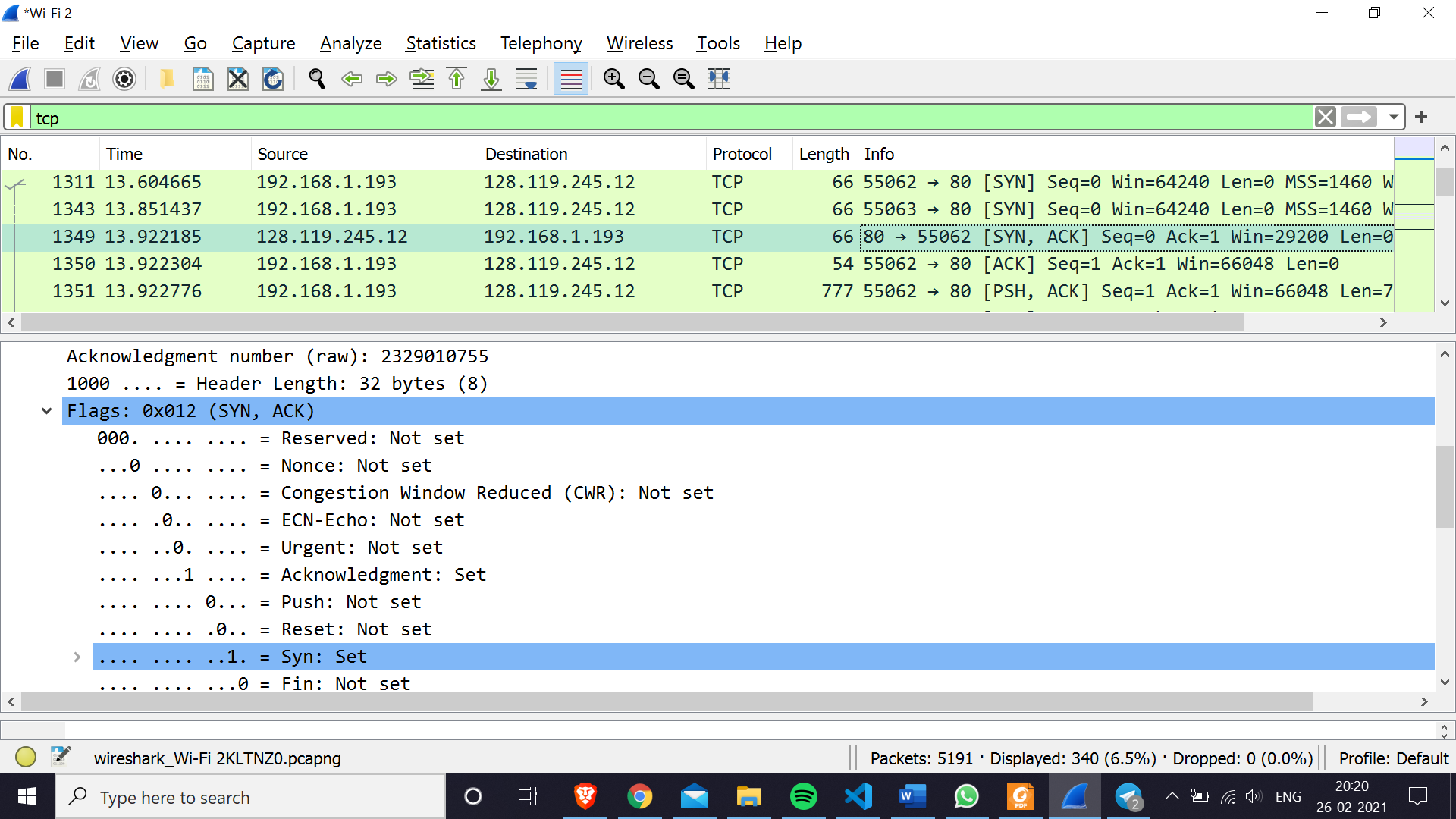
**Ans:** ACK = 1.

**c.** How did gaia.cs.umass.edu determine that value?

**Ans:** For a SYN message with sequence number X, the SYNACK message will response with acknowledgement number X + 1.

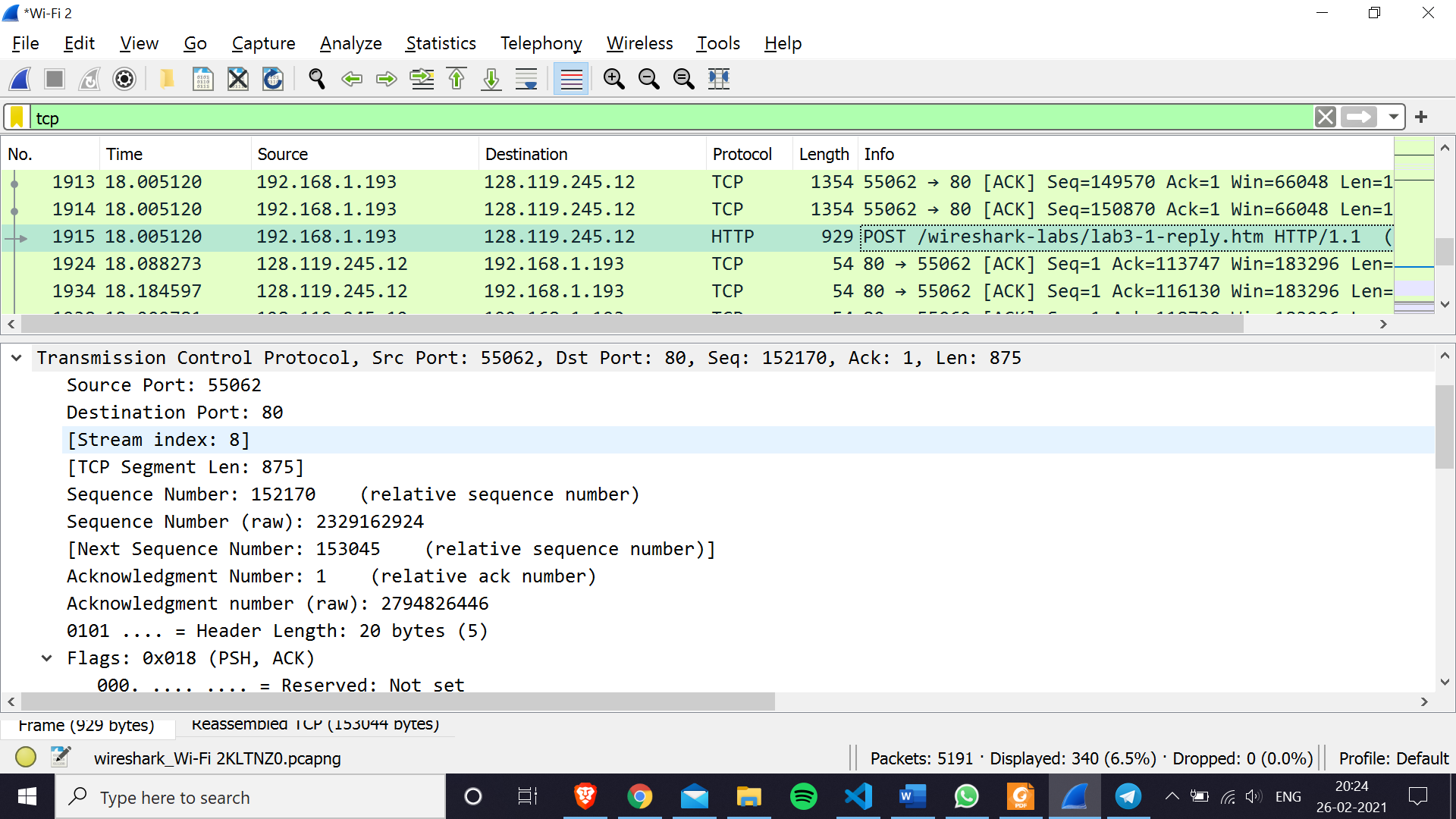
**d.** What is it in the segment that identifies the segment as a SYNACK segment?

**Ans:** Both SYN and ACK flags are set in the TCP part.



5. What is the sequence number of the TCP segment containing the HTTP POST command? Note that to find the POST command; you’ll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a “POST” within its DATA field.

Ans: Sequence number: 152170 (relative), 2329162924 (raw)



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

**Ans:** Sequence Number for Next 6 segments is

Seq Number = 1

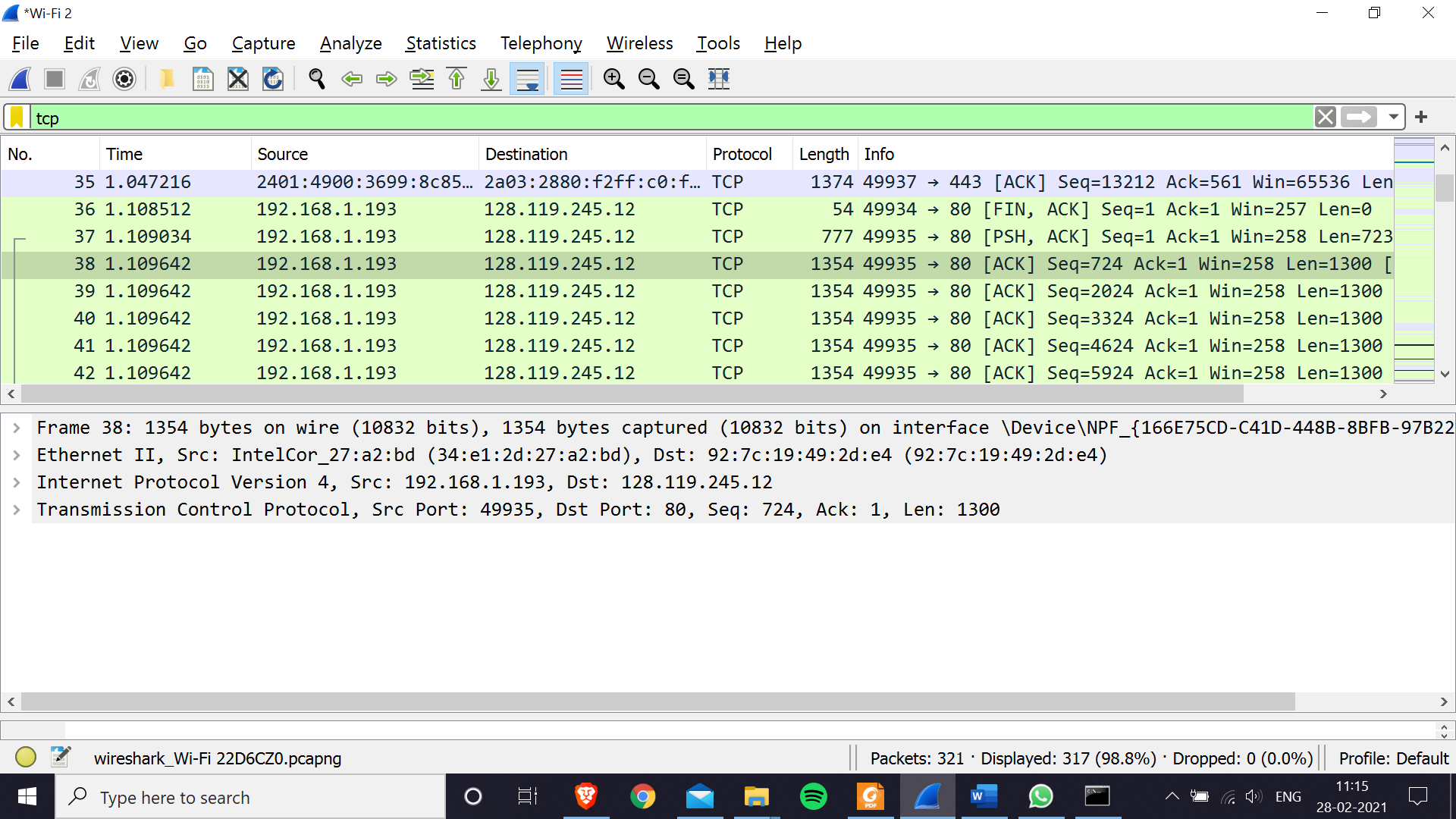
Seq Number = 724

Seq Number = 2024

Seq Number = 3324

Seq Number = 4624

Seq Number = 5924



**b.** At what time was each segment sent? And when was the ACK for each segment received?

**Ans:** Ack=1 Time sent and ACK received: 1.109034 1.725521

Ack =724 Time sent and ACK received: 1.109642 1.755166

Ack = 2024 Time sent and ACK received: 1.109642 1.765143

Ack = 3324 Time sent and ACK received: 1.109642 1.785378

Ack = 4624 Time sent and ACK received: 1.109642 1.807268

Ack = 5924 Time sent and ACK received: 1.109642 1.836426

**c.** What is the EstimatedRTT value after the receipt of each ACK? (\*Use the EstimatedRTT equation. 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).

**Ans:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Seq Number** | **Time** | **Received Time** | **Difference RTT** | **Estimated RTT** |
| 1 | 1.109034 | 1.725521 | 0.616487 | 0.616487 |
| 724 | 1.109642 | 1.755166 | 0.645524 | 0.620116 |
| 2024 | 1.109642 | 1.765143 | 0.655501 | 0.624539 |
| 3324 | 1.109642 | 1.785378 | 0.675736 | 0.630938 |
| 4624 | 1.109642 | 1.807268 | 0.697626 | 0.639273 |
| 5924 | 1.109642 | 1.836426 | 0.726784 | 0.650211 |

For First segment Estimated RTT= RTT

For Rest of the segment=0.125\*RTT(i)+0.875\*RTT(i-1)

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

**Ans:** 723,1300,1300,1300,1300,1300 (as seen in screenshot above).

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

**Ans:** First Non 1 ack packet: 3.855776

200 Ok status packets: 4.646113

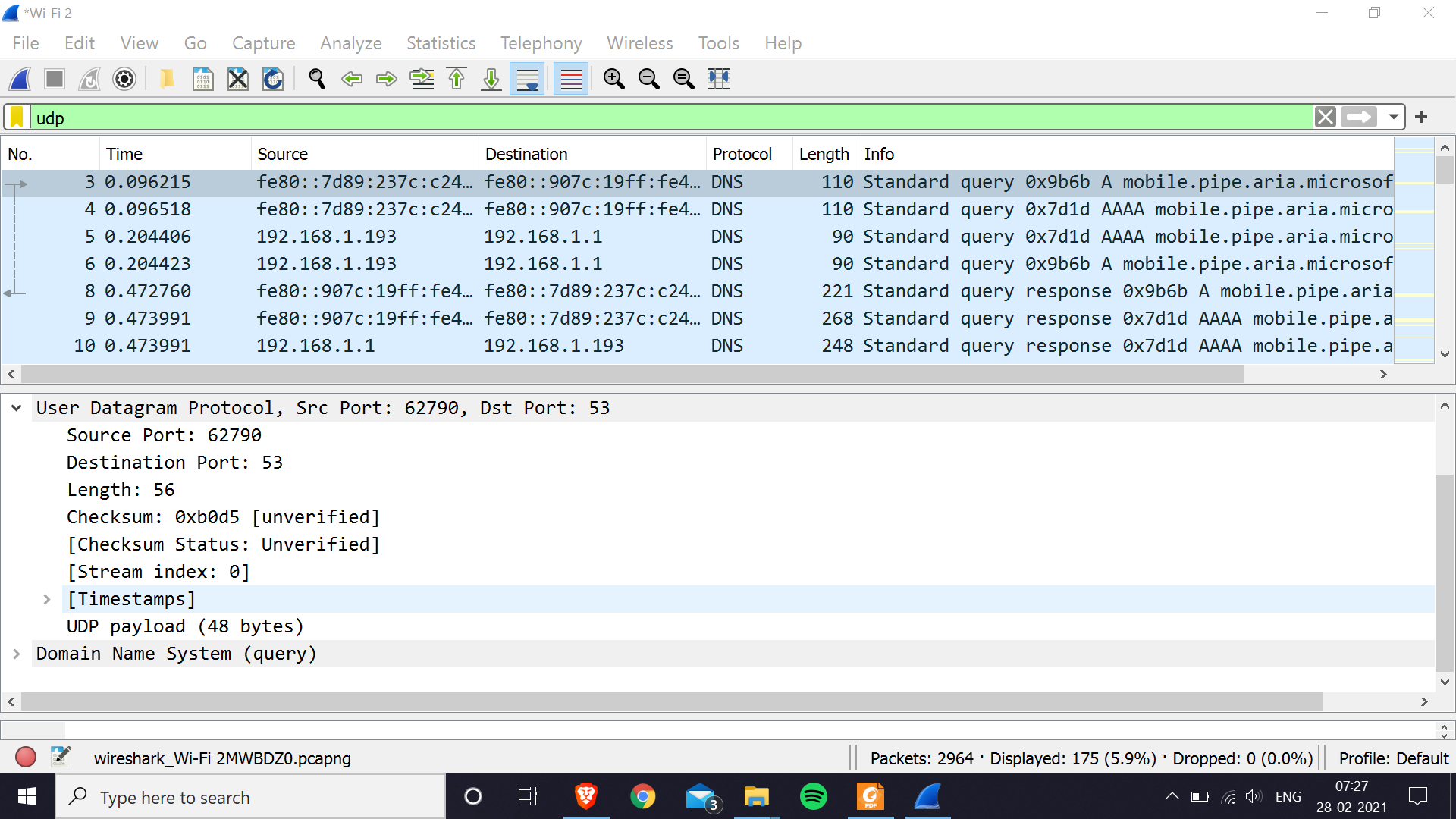
Throughput=Total Bytes (file size of alice.txt)/ Time Take (diff of above two values)

= 148 KB/0.790337

= **187.261889 KB/s**

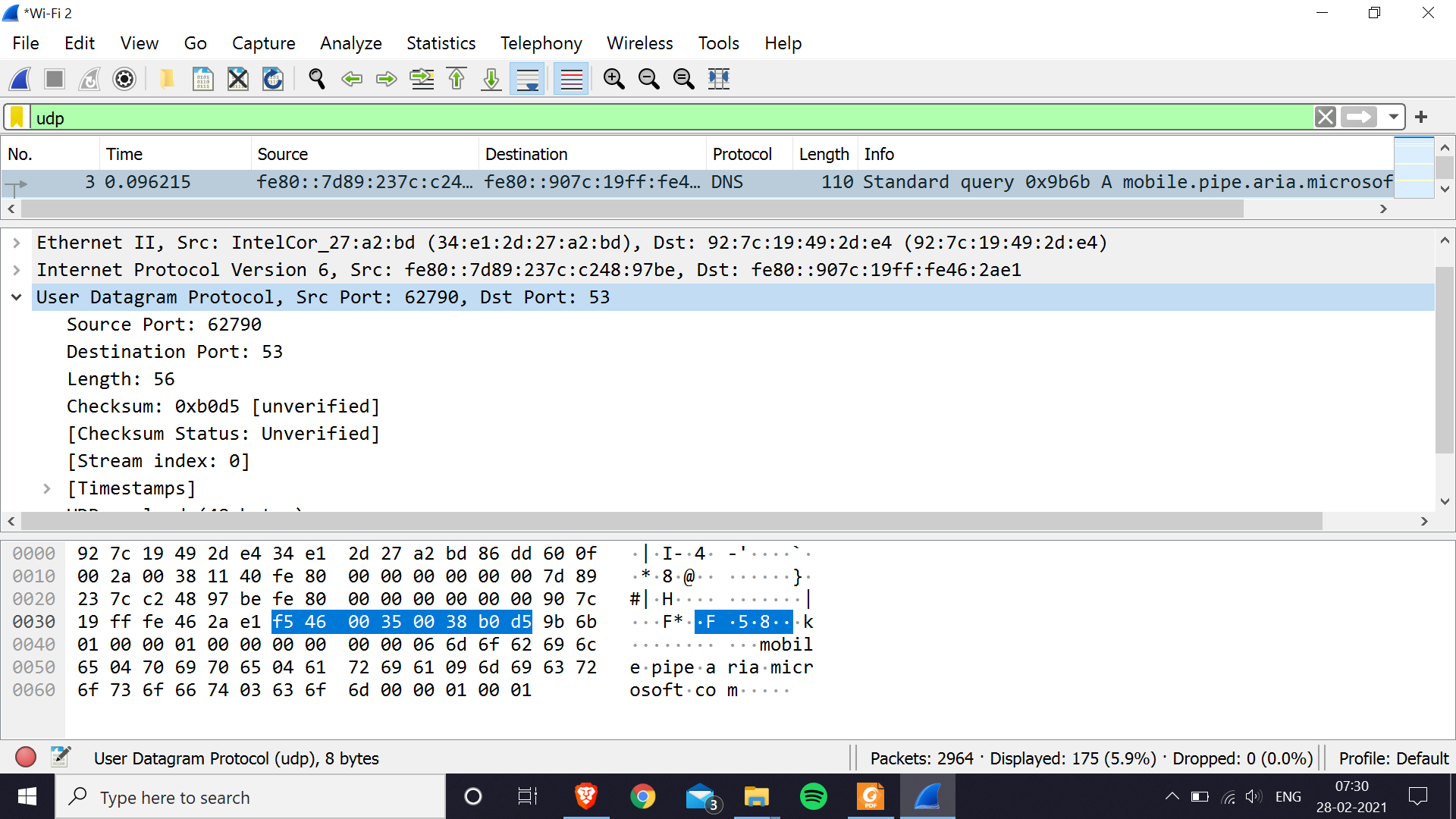
**1.** Select *one* UDP packet from your trace. From this packet, determine how many fields there are in the UDP header. Name these fields. (Answer these questions directly from what you observe in the packet trace.)

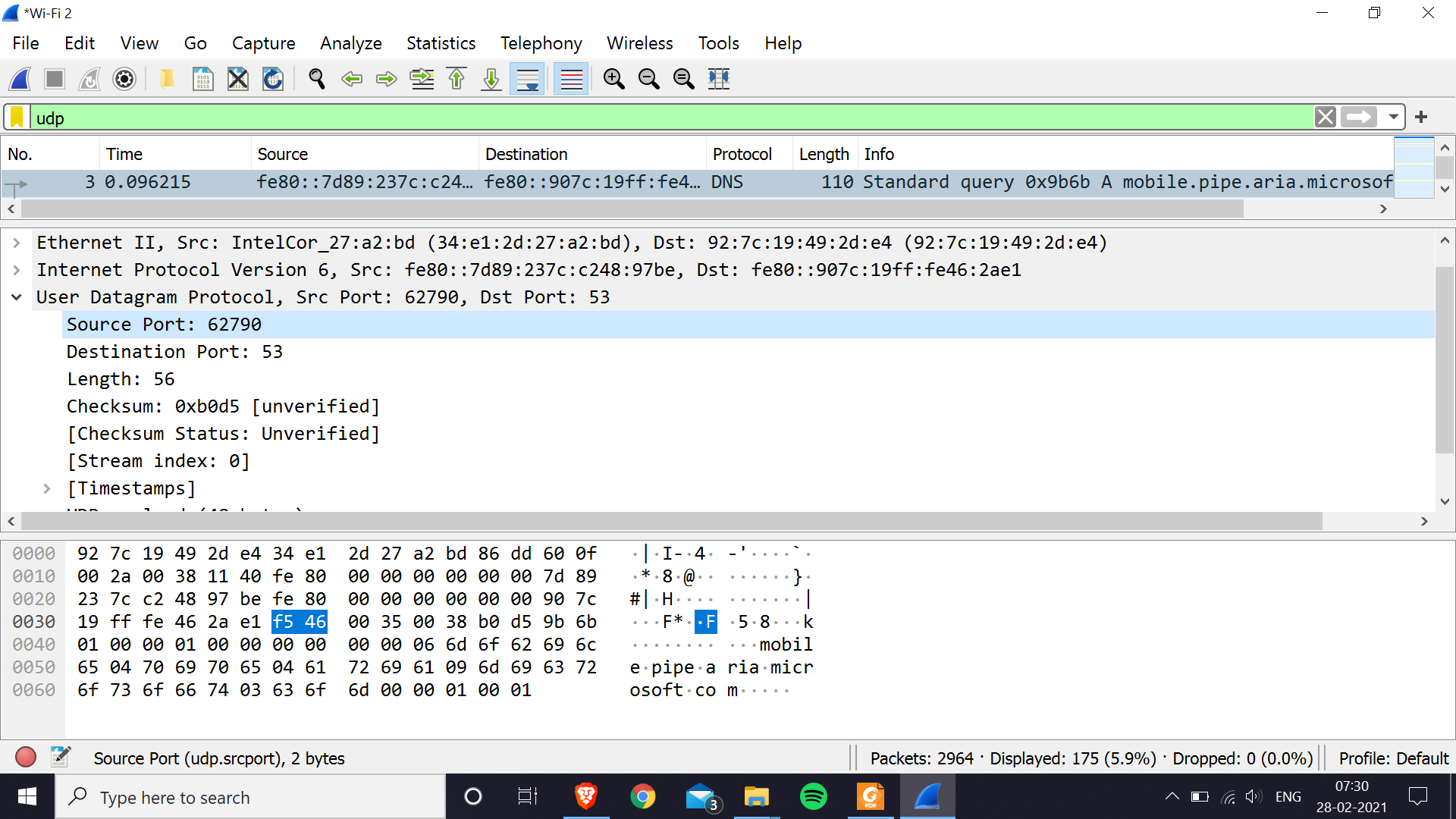
**Ans:** There are 4 header fields, namely Source Port, Destination Port, Length and Checksum.



**2.** By consulting the displayed information in Wireshark’s packet content field for this packet, determine the length (in bytes) of each of the UDP header fields.

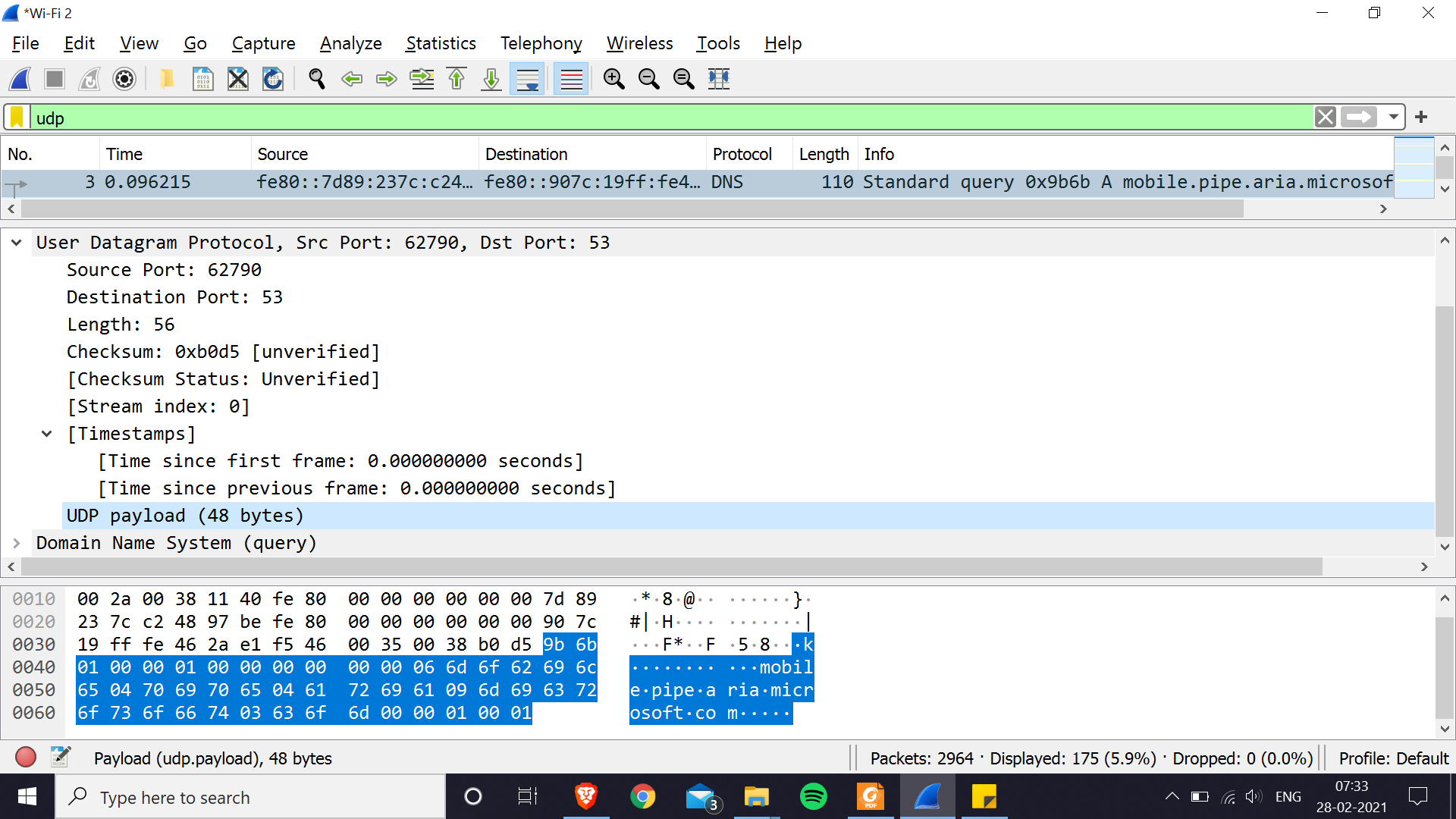
**Ans:** The UDP headers are occupying 8 bytes in total with 2 bytes being occupied by each of the 4 header fields (as seen in the below screenshots).





**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:** It is the total length of the UDP part (headers + data). Here the headers occupy 8 bytes and data is 48 bytes (see UDP payload in below screenshot) whose sum is 56 bytes as given in the length header field.



**4.** What is the maximum number of bytes that can be included in a UDP payload?

**Ans:** The maximum number of bytes that can be in the payload is 2^16- the bytes already being used by the header field (8). Therefore, the maximum payload is 65535-8= 65527 bytes.

**5.** What is the largest possible source port number?

**Ans:** The largest possible source port number is 2^16 or 65535.

**6.** What is the protocol number for UDP?

**Ans:** It is 17 (as seen in below screenshot).

