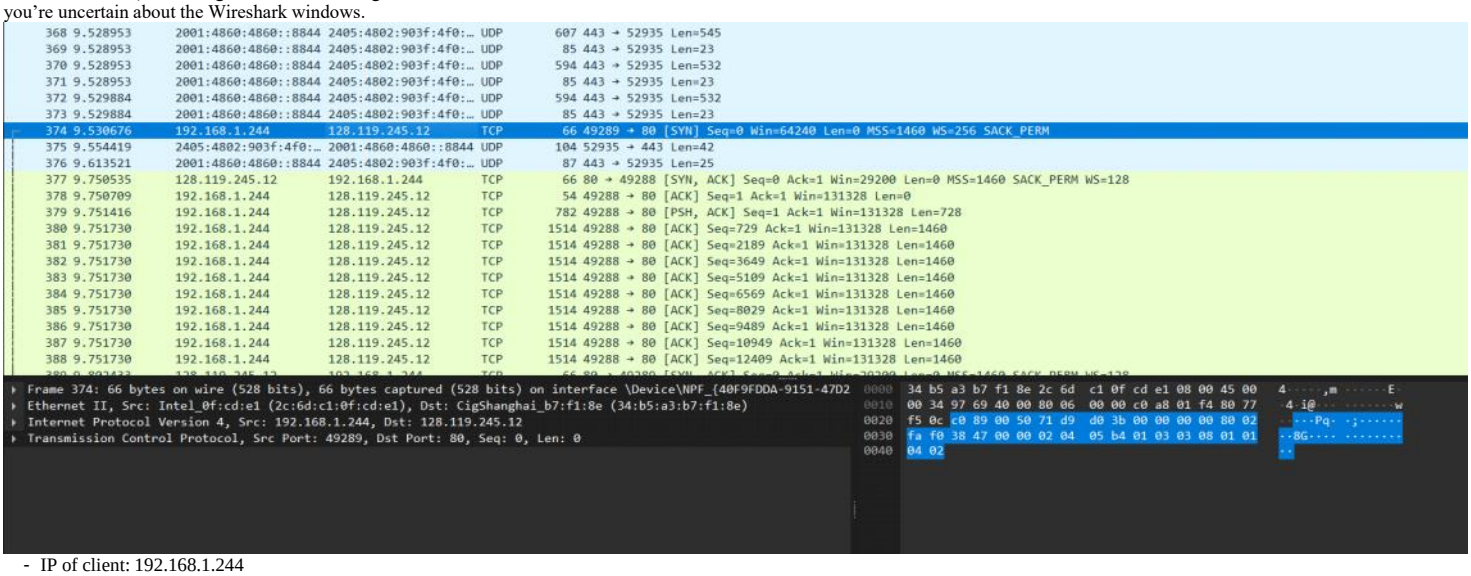
Lab 4a - TCP

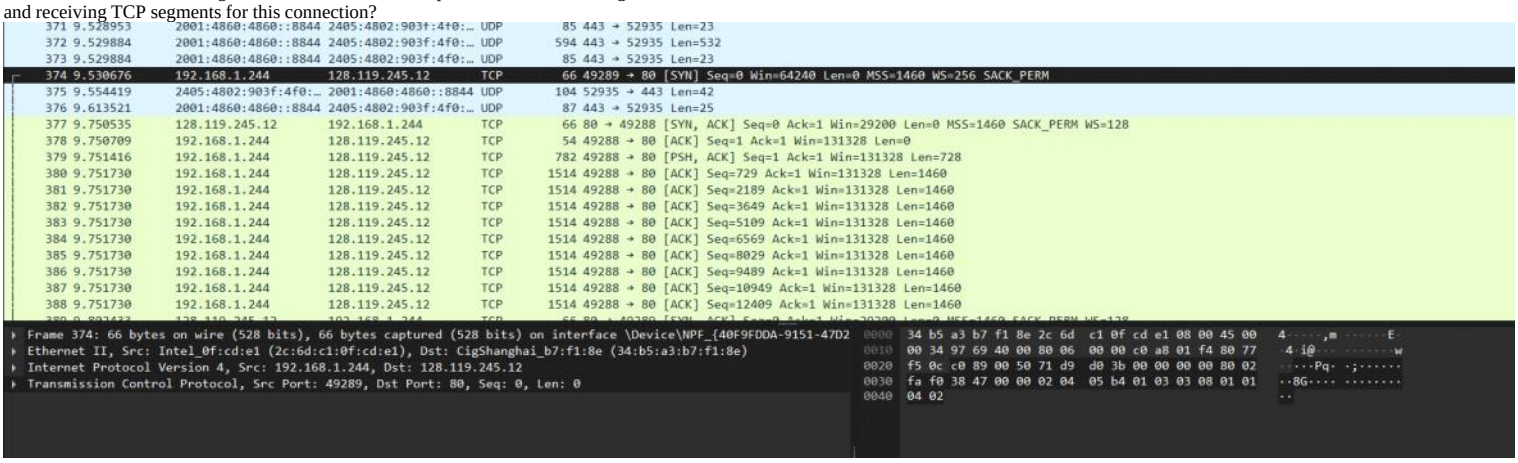
Wednesday, October 23, 2024 11:28 PM

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? 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.



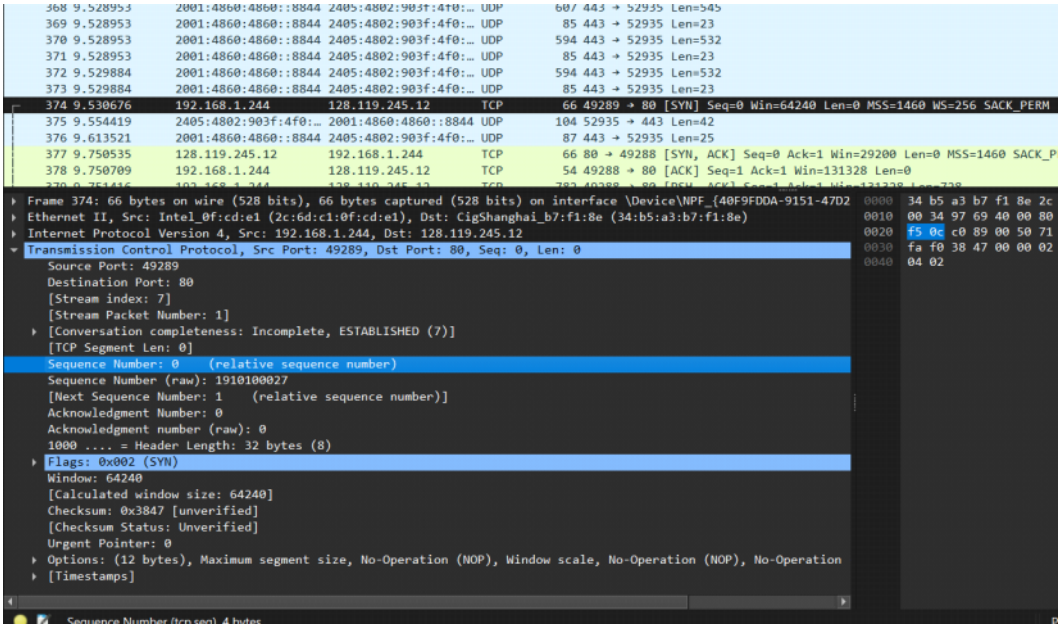
- IP of client: 192.168.1.244
- TCP Port of client: 49288

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?



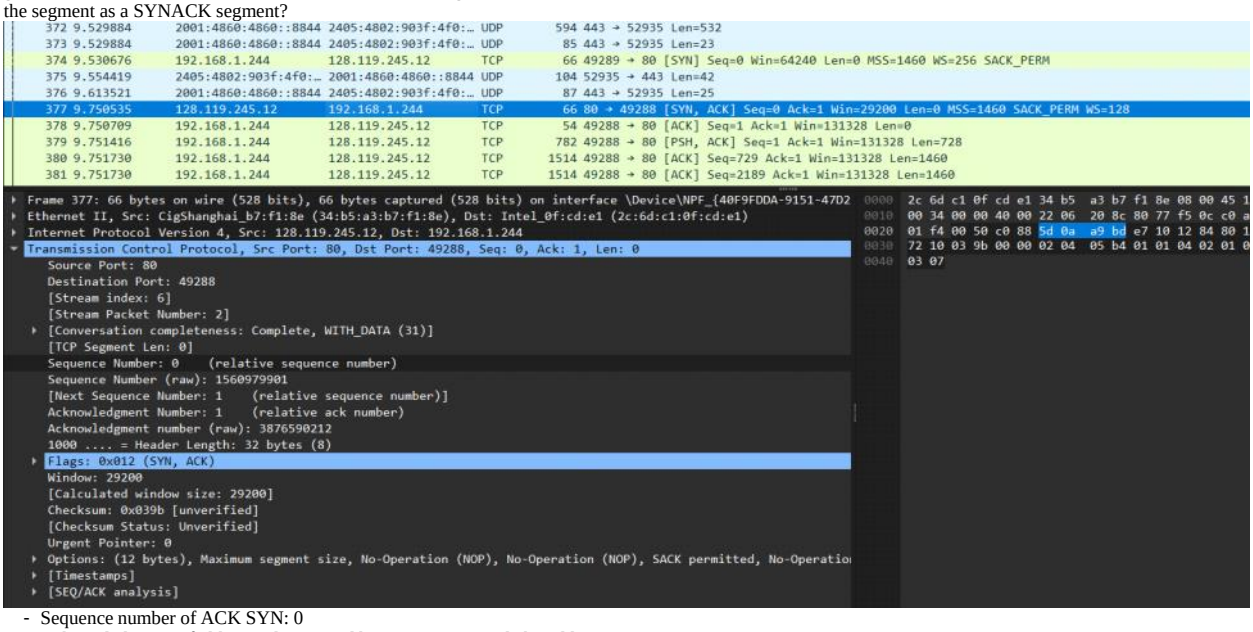
- IP of gaia.cs.umass.edu: 128.119.245.12
- TCP port for send and receive: 80

4. 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?



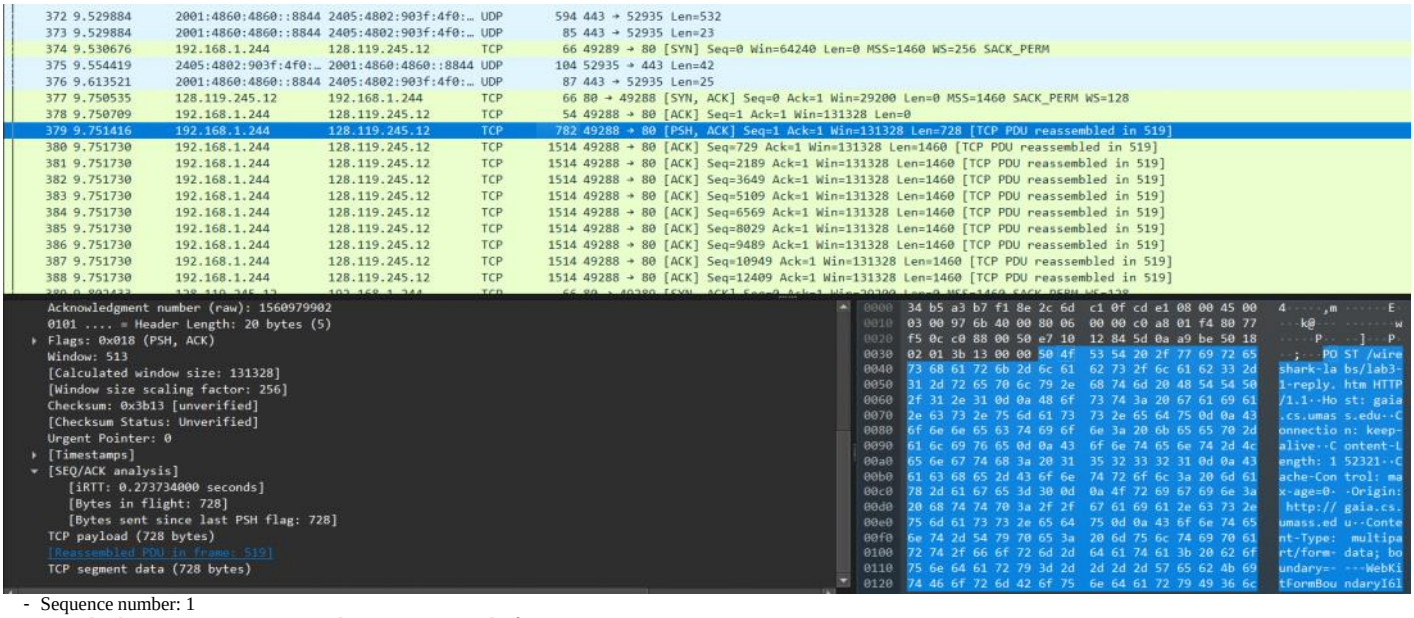
- Sequence number: 0
- SYN segment: Flage: 0x002 (SYN)

5. 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?



- Sequence number of ACK SYN: 0
- Acknowledgement field: 1, is determined by gaia.cs.umass.edu by adding 1 to the initial sequence number of SYN segment from the client computer
- Identifies the segment as a SYNACK segment: Flags 0x012 (SYN,ACK)

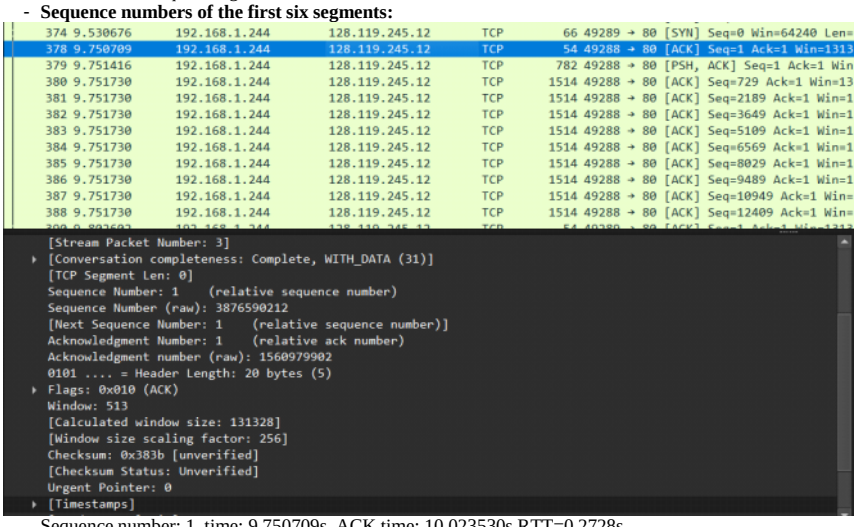
6. What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.



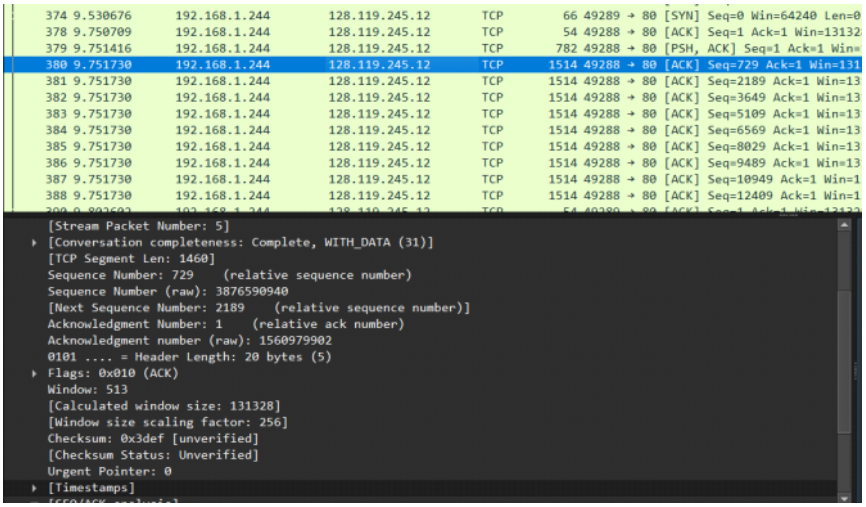
- Sequence number: 1

7. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? 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? What is the EstimatedRTT value (see Section 3.5.3, page 242 in text) after the receipt of each ACK? 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 on page 242 for all subsequent segments

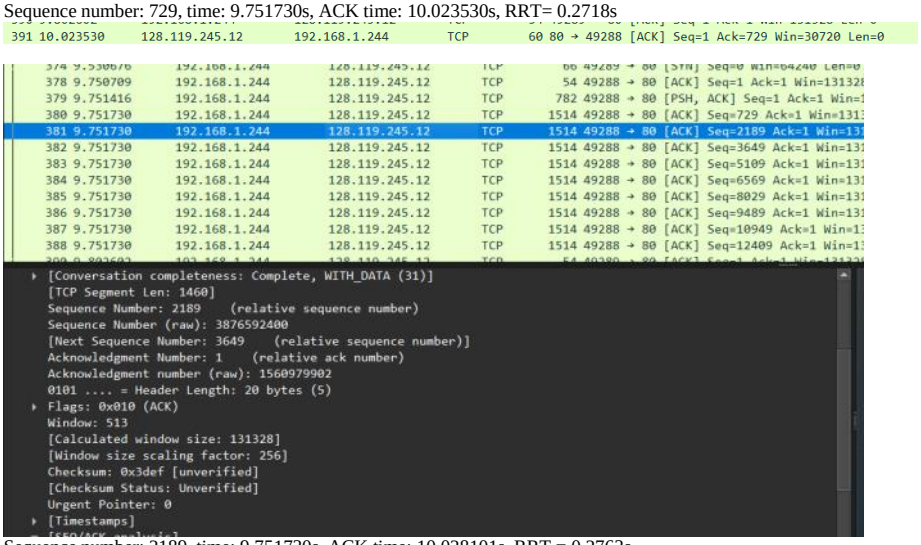
- **Sequence numbers of the first six segments:**



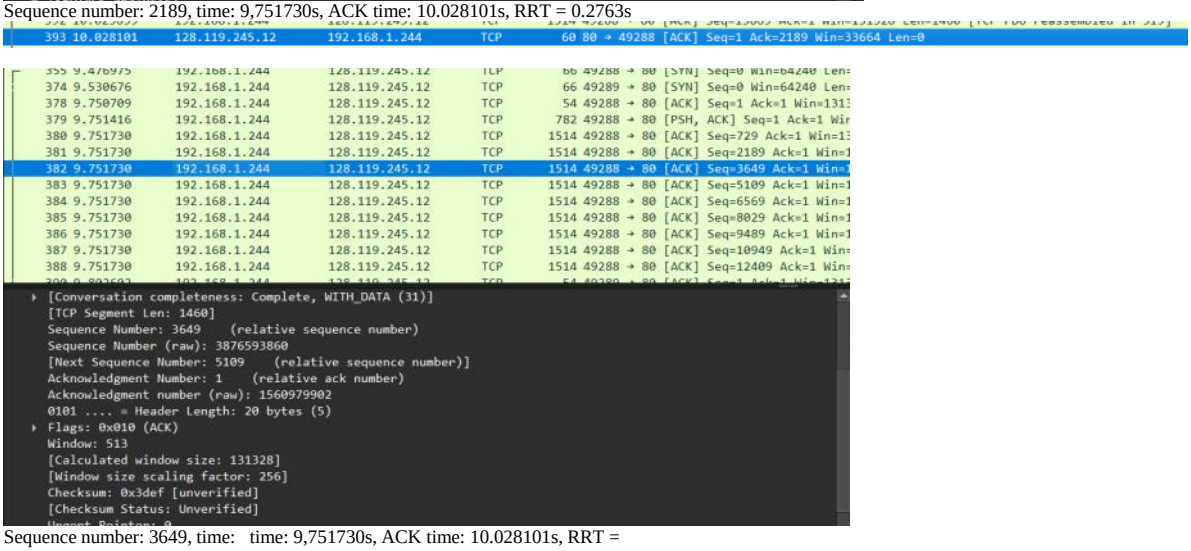
- Sequence number: 1, time: 9.750709s, ACK time: 10.023530s,RTT=0.2728s



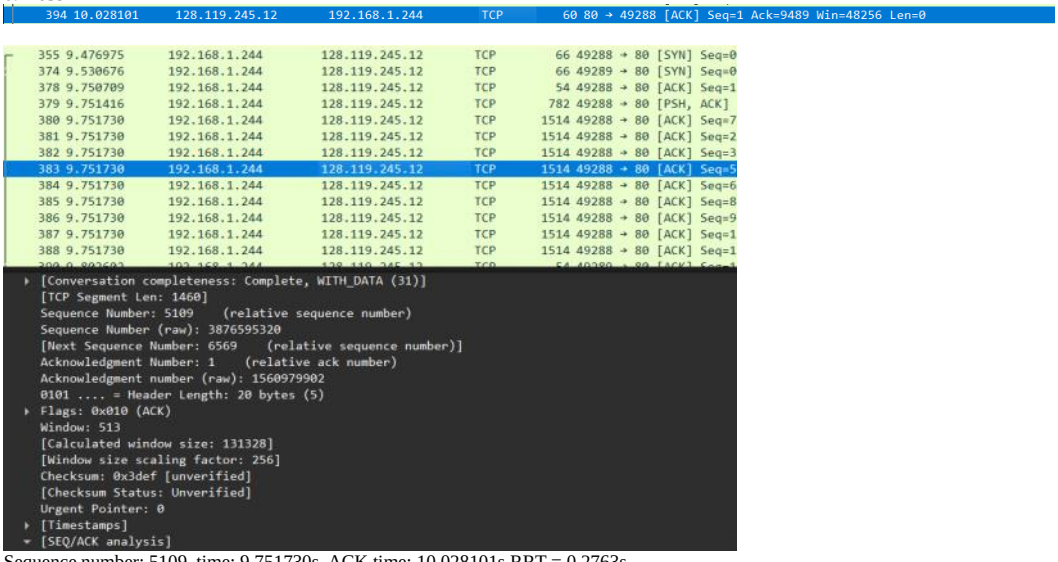
- Sequence number: 729, time: 9.751730s, ACK time: 10.023530s, RRT= 0.2718s



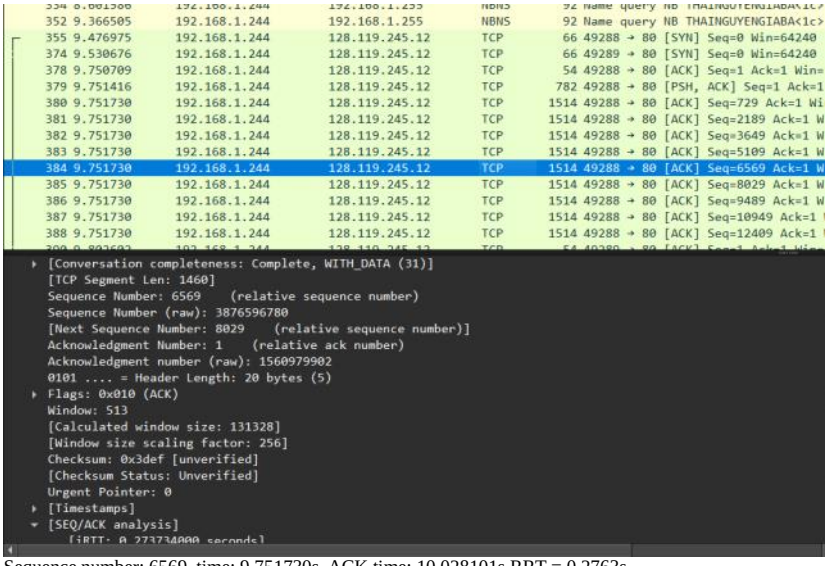
- Sequence number: 2189, time: 9,751730s, ACK time: 10.028101s, RRT = 0.2763s



- Sequence number: 3649, time: time: 9,751730s, ACK time: 10.028101s, RRT = 0.2763s

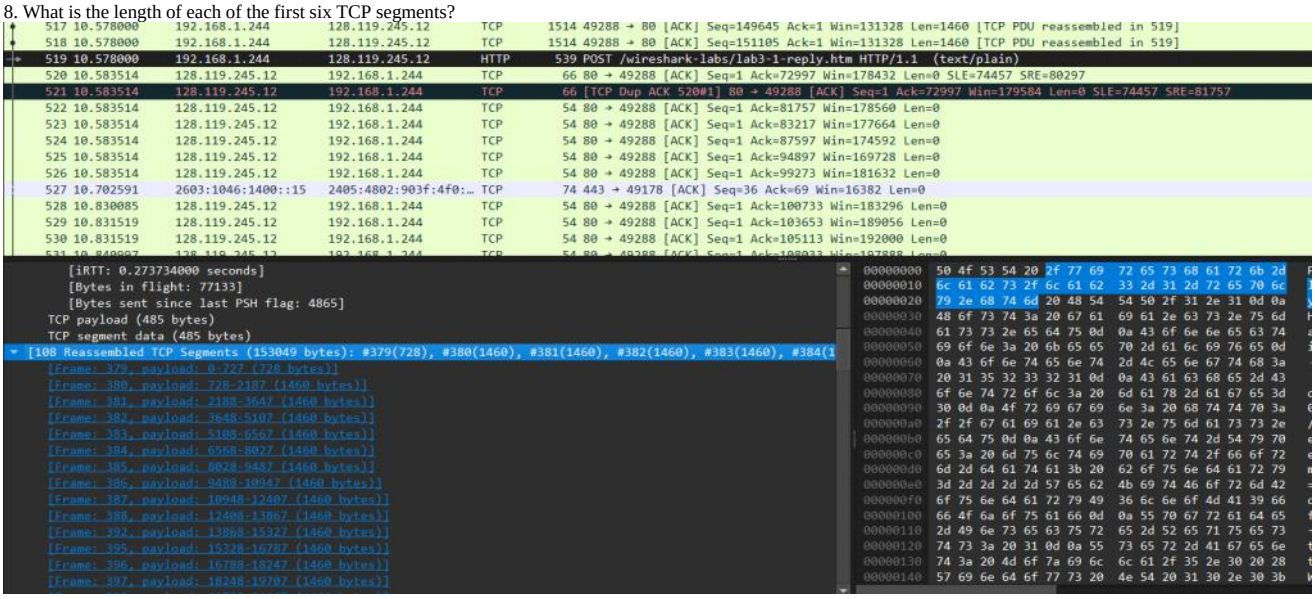


- Sequence number: 5109, time: 9,751730s, ACK time: 10.028101s,RRT = 0.2763s



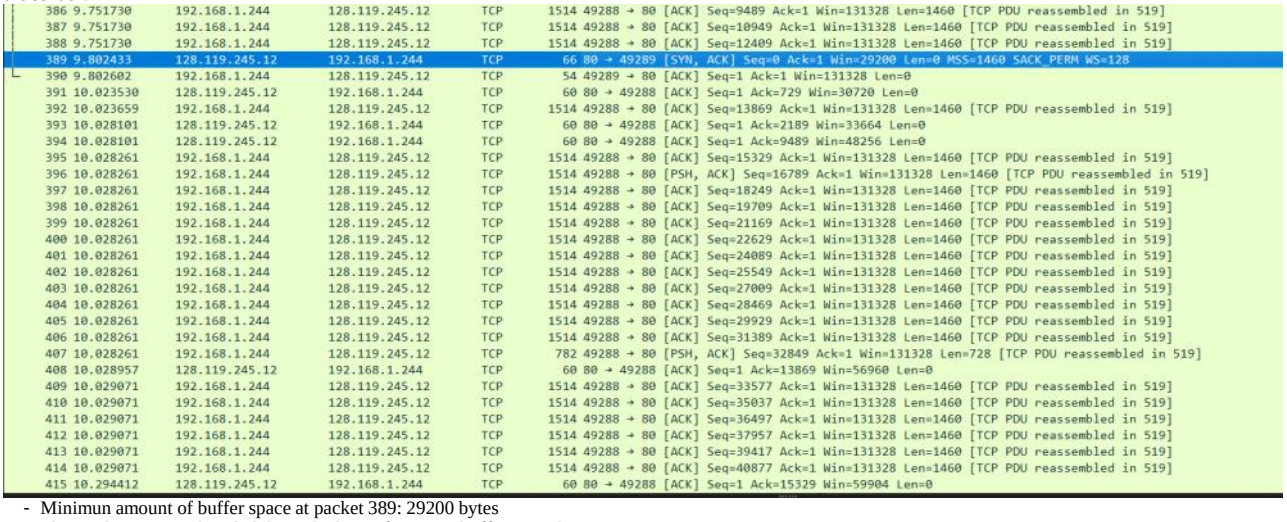
- Sequence number: 6569, time: 9,751730s ,ACK time: 10.028101s,RRT = 0.2763s

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



- First segment: 728 bytes
- Other five segment: 1460 bytes

9. What is 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?

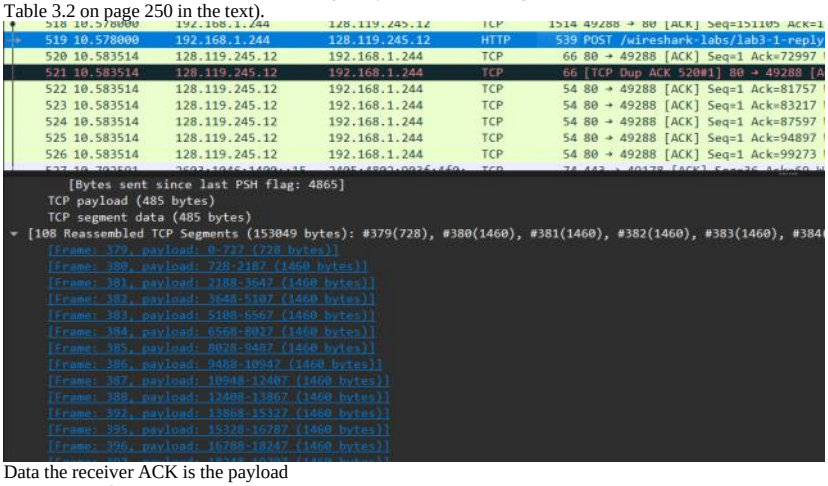


- Minimun amount of buffer space at packet 389: 29200 bytes
- The sender is never throttled due to lacking of receiver buffer space by

inspecting this trace.

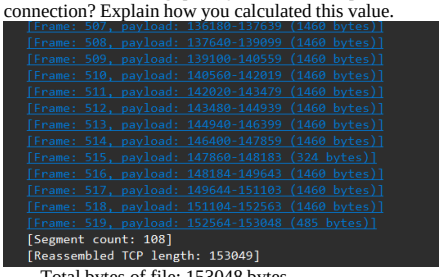
10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?

11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 250 in the text).

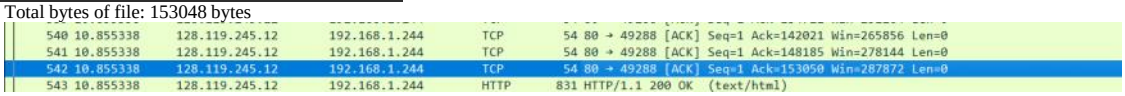


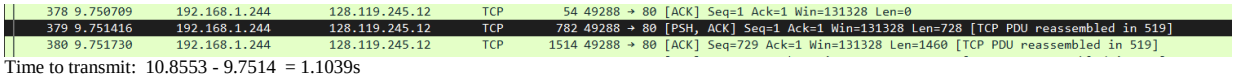
- Data the receiver ACK is the payload
  - 727 bytes
  - 2187-728 bytes
  - …

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



- Total bytes of file: 153048 bytes

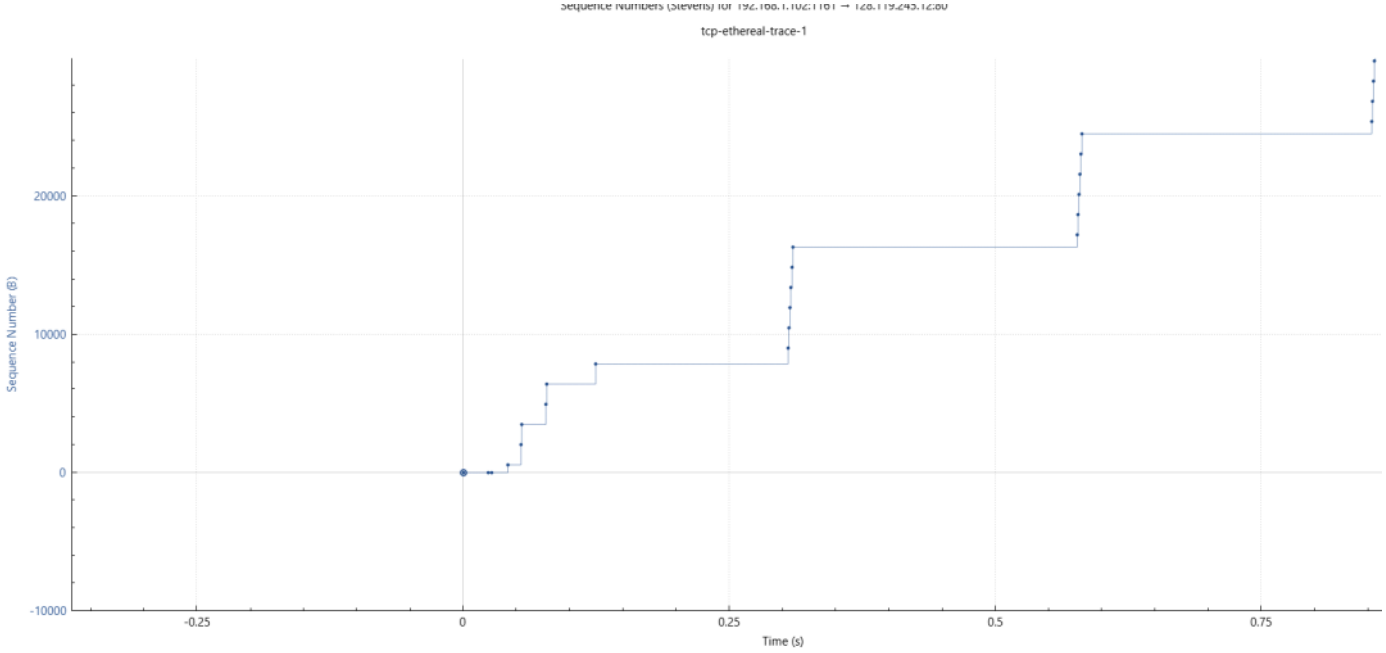




- Time to transmit: 10.8553 - 9.7514 = 1.1039s

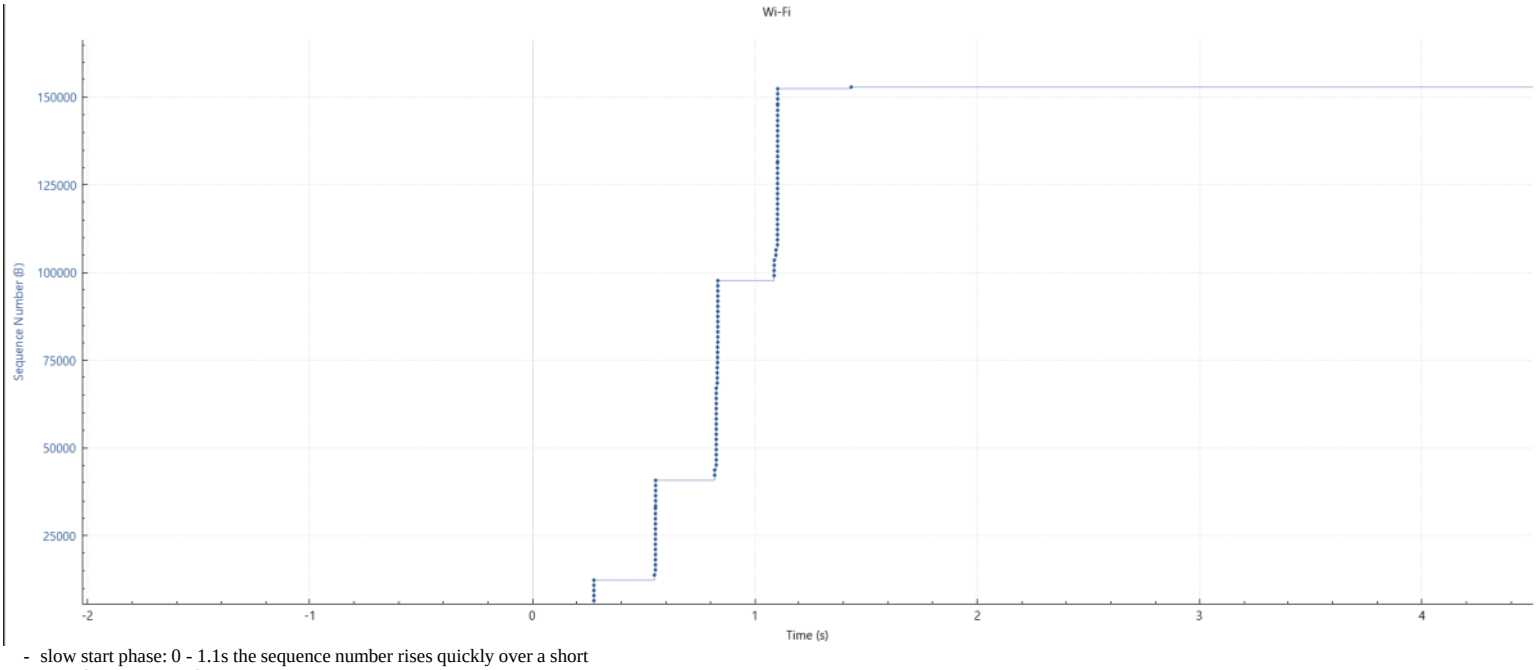
=> Throughput: 153048 /1.1039 = 138642.993 Byte/s = 135.3926 KB/s

13. 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 TCP's slowstart phase begins and ends, and where congestion avoidance takes over? Comment on ways in which the measured data differs from the idealized behavior of TCP that we've studied in the text.



- slowstart phase (0 - 0.3s): the sequence number rises quickly over a short period
- congestion avoidance (0.3 - inf) the slope of the sequence number increase becomes less steep

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



- slow start phase: 0 - 1.1s the sequence number rises quickly over a short period (steep upward steps).
- congestion avoidance: 1.1s - 4.4s the slope of the sequence number increase becomes less steep

Lab 5b - DHCP lab

Thursday, October 24, 2024 3:13 PM

1. Are DHCP messages sent over UDP or TCP?

No.	Time	Source	Destination	Protocol	Length	Info
28	1.827229	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0xafa2140c
34	2.343502	0.0.0.0	255.255.255.255	DHCP	352	DHCP Request - Transaction ID 0xafa2140c
212	10.673606	0.0.0.0	255.255.255.255	DHCP	346	DHCP Request - Transaction ID 0x899ba273
271	12.787544	0.0.0.0	255.255.255.255	DHCP	346	DHCP Request - Transaction ID 0x899ba273
302	14.301297	191.16.11.129	191.16.1.2	DHCP	342	DHCP Release - Transaction ID 0x8e4f28a9
344	20.077021	0.0.0.0	255.255.255.255	DHCP	345	DHCP Discover - Transaction ID 0xdeed530d
379	21.080930	191.16.1.2	191.16.11.129	DHCP	342	DHCP Offer - Transaction ID 0xdeed530d
381	21.083443	0.0.0.0	255.255.255.255	DHCP	372	DHCP Request - Transaction ID 0xdeed530d
382	21.091061	191.16.1.2	191.16.11.129	DHCP	342	DHCP ACK - Transaction ID 0xdeed530d
415	21.392952	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x825781d7
443	21.816303	0.0.0.0	255.255.255.255	DHCP	346	DHCP Request - Transaction ID 0x34d708ab
857	23.126717	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0x825781d7
1593	26.485541	191.16.11.129	191.16.1.2	DHCP	360	DHCP Request - Transaction ID 0xa1849ec8
1606	26.490328	191.16.1.2	191.16.11.129	DHCP	342	DHCP ACK - Transaction ID 0xa1849ec8
2345	38.069962	191.16.11.129	191.16.1.2	DHCP	342	DHCP Release - Transaction ID 0x45d1c3a3
2406	46.242941	0.0.0.0	255.255.255.255	DHCP	345	DHCP Discover - Transaction ID 0x1efc5691
2407	46.383644	191.16.1.2	191.16.11.129	DHCP	342	DHCP Offer - Transaction ID 0x1efc5691
2408	46.385933	0.0.0.0	255.255.255.255	DHCP	372	DHCP Request - Transaction ID 0x1efc5691
2409	46.402067	191.16.1.2	191.16.11.129	DHCP	342	DHCP ACK - Transaction ID 0x1efc5691

```
▸ Frame 28: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits) on interface \Device\NPF_{40F9FDDA-9151-4
▸ Ethernet II, Src: Intel_5f:09:33 (88:d8:2e:5f:09:33), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Internet Protocol Version 4, Src: 0.0.0.0, Dst: 255.255.255.255
▾ User Datagram Protocol, Src Port: 68, Dst Port: 67
    Source Port: 68
    Destination Port: 67
    Length: 308
    Checksum: 0xd50d [unverified]
    [Checksum Status: Unverified]
    [Stream index: 6]
    [Stream Packet Number: 1]
  ▸ [Timestamps]
    UDP payload (300 bytes)
▸ Dynamic Host Configuration Protocol (Discover)
```

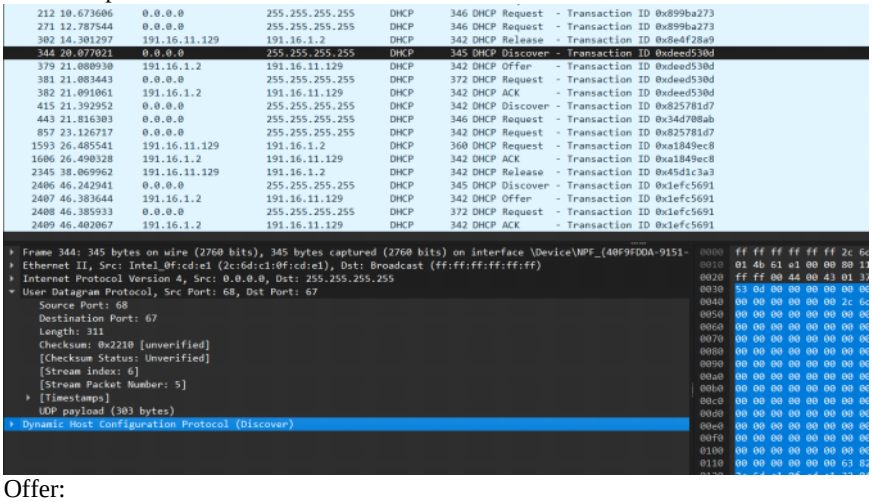
- Send over UDP

2. Draw a timing datagram illustrating the sequence of the first four-packet Discover/Offer/Request/ACK DHCP exchange between the client and server. For each packet, indicated the source and destination port numbers. Are the port numbers the same as in the example given in this lab assignment?

- Timing datagram:

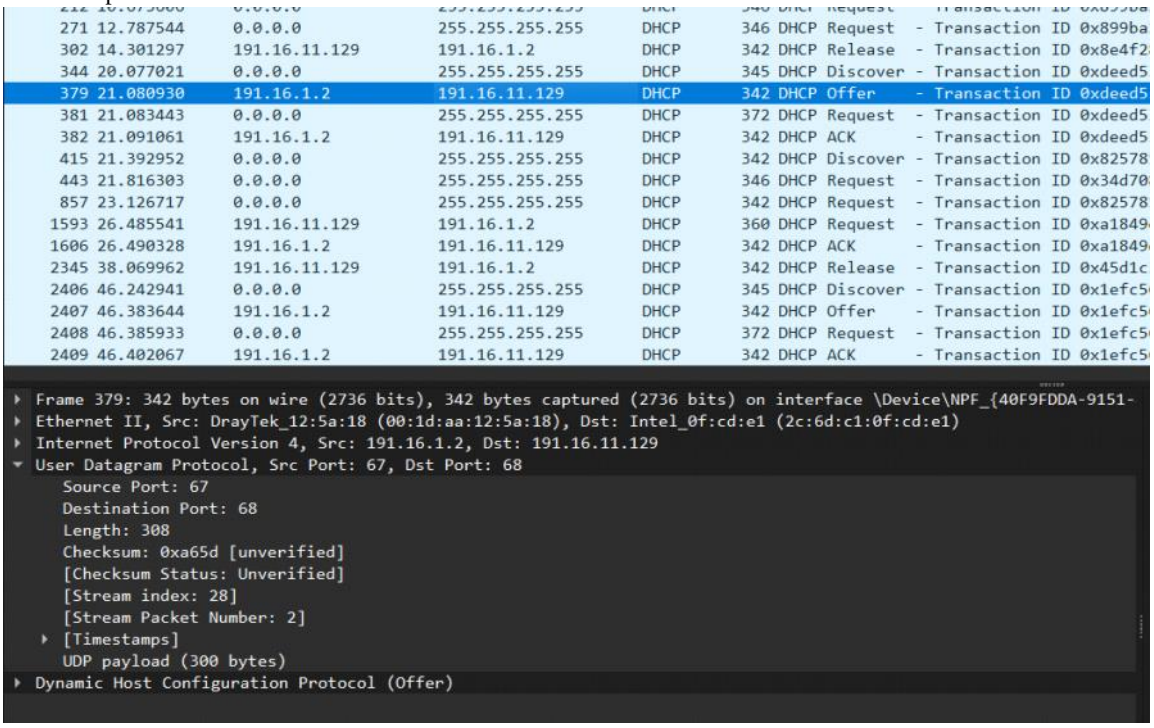
344	20.077021	0.0.0.0	255.255.255.255	DHCP	345	DHCP Discover - Transaction ID 0xdeed530d
379	21.080930	191.16.1.2	191.16.11.129	DHCP	342	DHCP Offer - Transaction ID 0xdeed530d
381	21.083443	0.0.0.0	255.255.255.255	DHCP	372	DHCP Request - Transaction ID 0xdeed530d
382	21.091061	191.16.1.2	191.16.11.129	DHCP	342	DHCP ACK - Transaction ID 0xdeed530d

- Discover:
  - Source port: 68
  - Dest port: 67

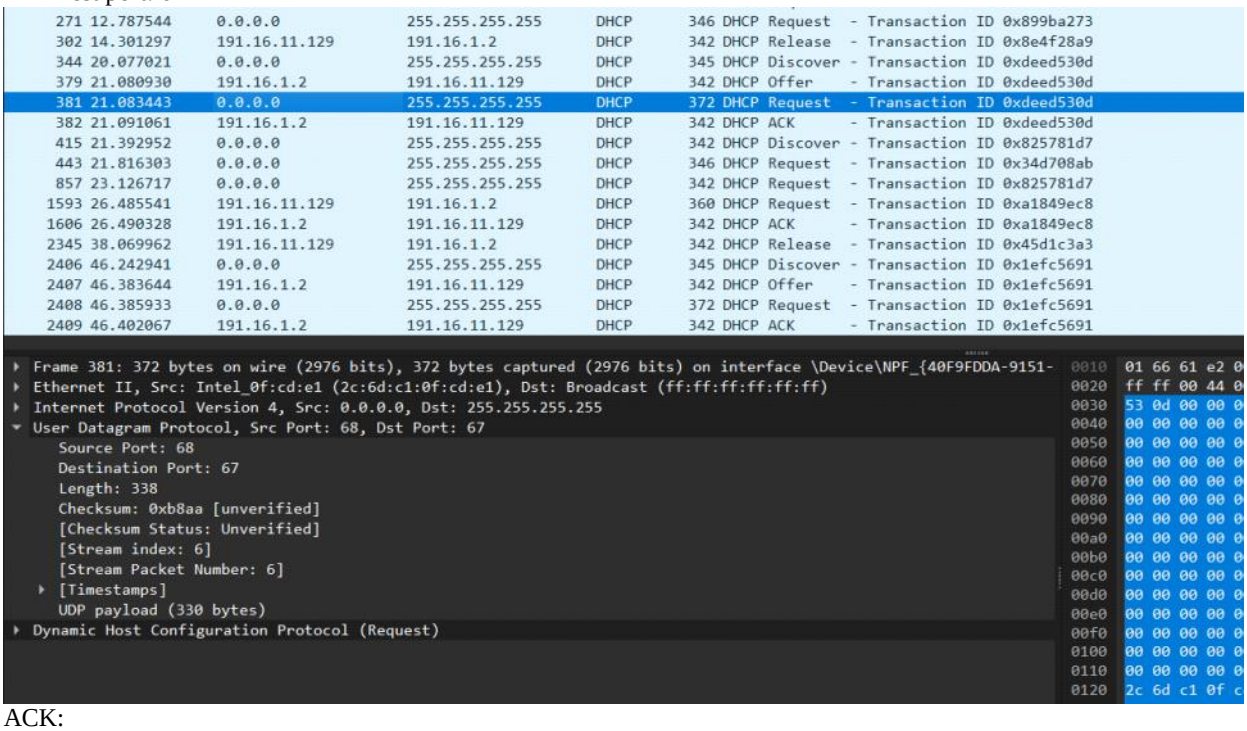


- Offer:

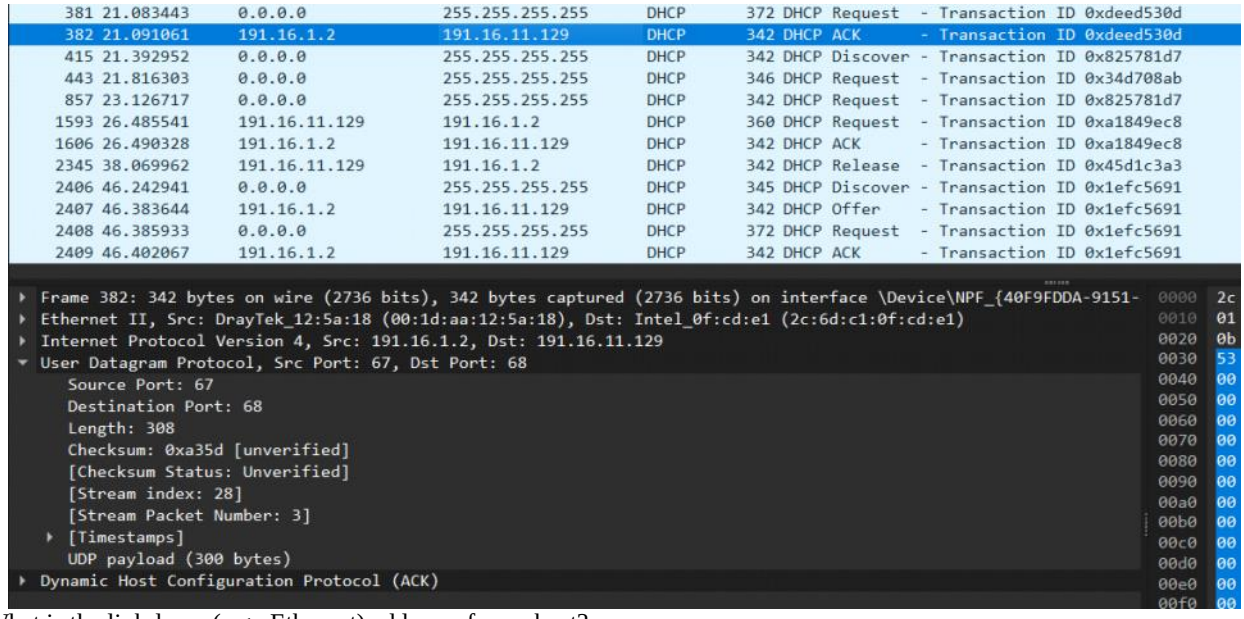
- Source port: 67
- Dest port: 68



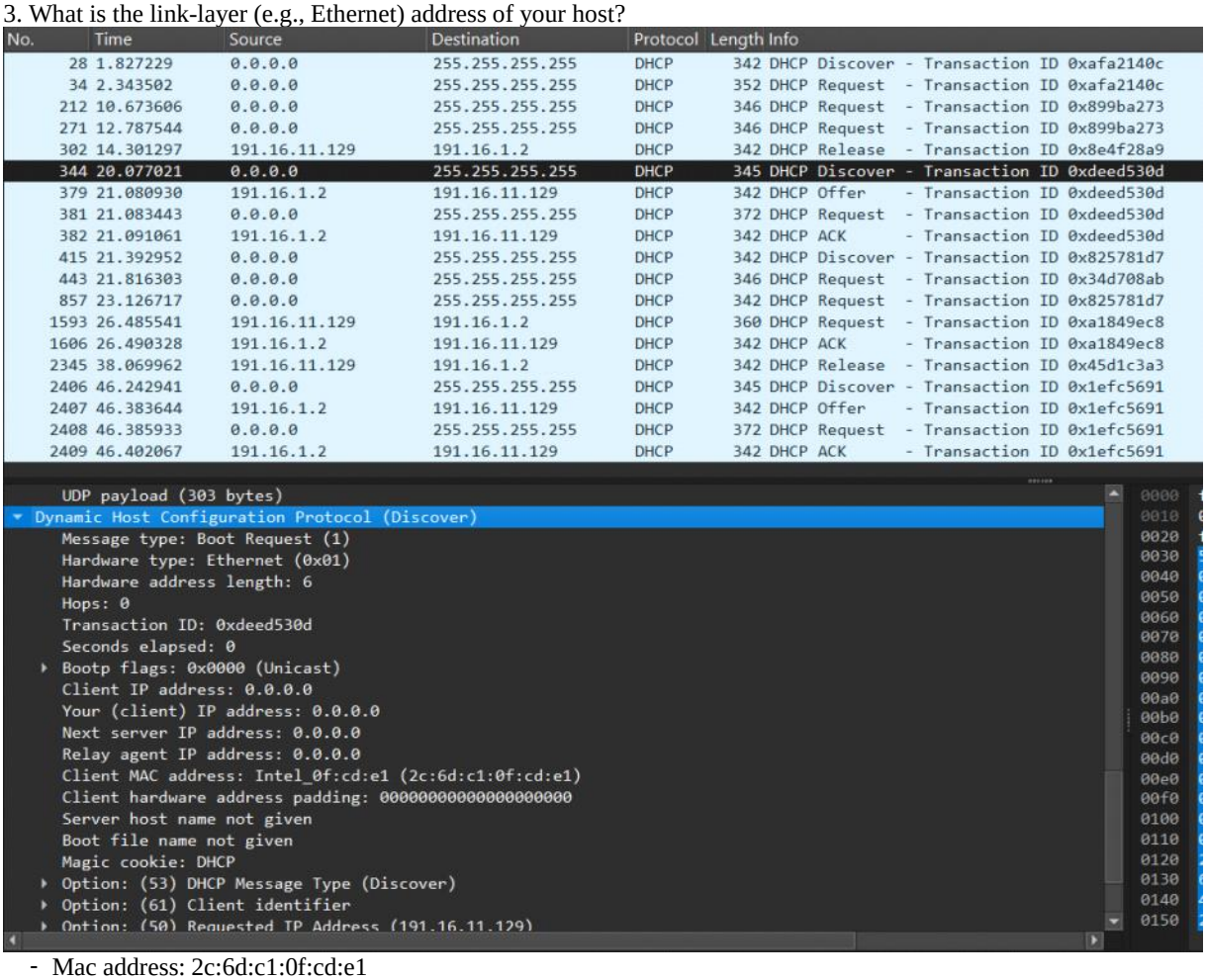
- Request:
  - Source port: 68
  - Dest port: 67



- ACK:
  - Source port: 67
  - Dest port: 68



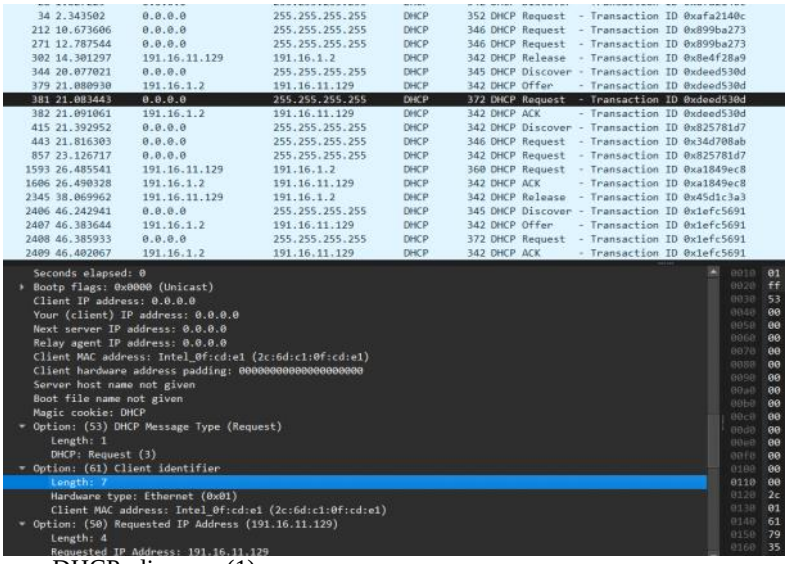
3. What is the link-layer (e.g., Ethernet) address of your host?



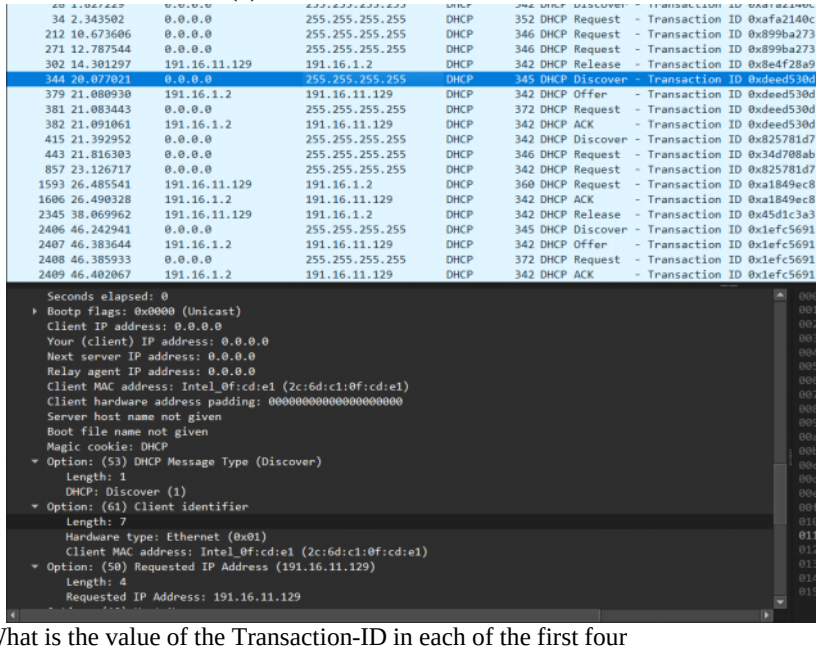
- Mac address: 2c:6d:c1:0f:cd:e1

4. What values in the DHCP discover message differentiate this message from the DHCP request message?

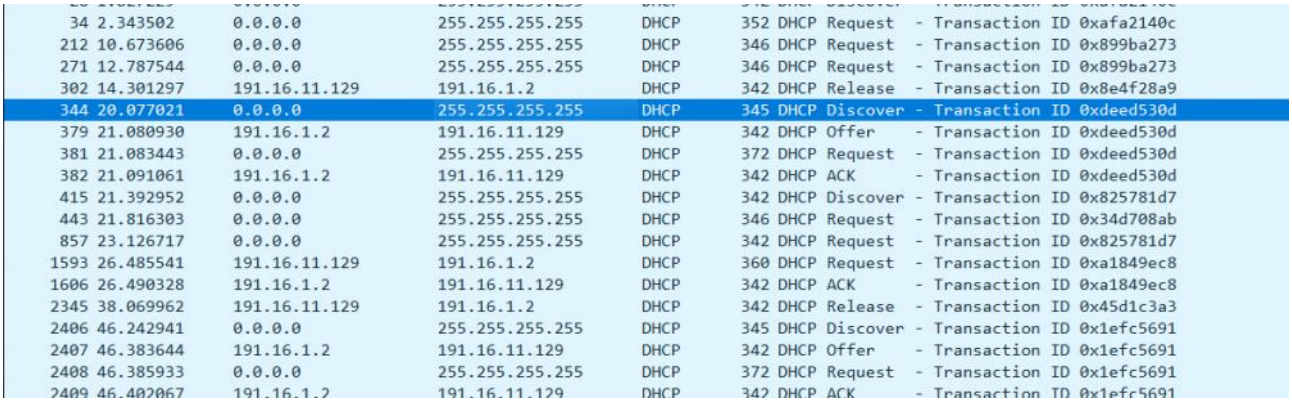
- Option 53:
  - DHCP: request (3)



- DHCP: discover (1)



5. What is the value of the Transaction-ID in each of the first four (Discover/Offer/Request/ACK) DHCP messages? What are the values of the Transaction-ID in the second set (Request/ACK) set of DHCP messages? What is the purpose of the Transaction-ID field?



- First four Transaction-ID: 0xdeed530d
- Second four Transaction-ID: 0x1efc5691 (I get the third one as a second one)
- Purpose: enabling the client to identify the related dhcp answer to each request

6. A host uses DHCP to obtain an IP address, among other things. But a host's IP address is not confirmed until the end of the four-message exchange! If the IP

address is not set until the end of the four-message exchange, then what values are used in the IP datagrams in the four-message exchange? For each of the four DHCP messages (Discover/Offer/Request/ACK DHCP), indicate the source and destination IP addresses that are carried in the encapsulating IP datagram.

```
344 20.077021    0.0.0.0          255.255.255.255    DHCP    345 DHCP Discover - Transaction ID 0xdeed530d
379 21.080930    191.16.1.2       191.16.11.129      DHCP    342 DHCP Offer    - Transaction ID 0xdeed530d
381 21.083443    0.0.0.0          255.255.255.255    DHCP    372 DHCP Request  - Transaction ID 0xdeed530d
382 21.091061    191.16.1.2       191.16.11.129      DHCP    342 DHCP ACK      - Transaction ID 0xdeed530d
```

	Src IP	Dest IP
Discover	0.0.0.0	255.255.255.255
- Offer	191.16.1.2	191.16.11.129
Request	0.0.0.0	255.255.255.255
ACK	191.16.1.2	191.16.11.129

7. What is the IP address of your DHCP server?
   - IP address of DHCP server: 191.16.1.2
8. What IP address is the DHCP server offering to your host in the DHCP Offer message? Indicate which DHCP message contains the offered DHCP address.
   - IP address offer: 191.16.11.129

```
302 14.301297    191.16.11.129    191.16.1.2         DHCP    342 DHCP Release  - Transaction ID 0x8e4f28a9
344 20.077021    0.0.0.0          255.255.255.255    DHCP    345 DHCP Discover - Transaction ID 0xdeed530d
379 21.080930    191.16.1.2       191.16.11.129      DHCP    342 DHCP Offer    - Transaction ID 0xdeed530d
381 21.083443    0.0.0.0          255.255.255.255    DHCP    372 DHCP Request  - Transaction ID 0xdeed530d
382 21.091061    191.16.1.2       191.16.11.129      DHCP    342 DHCP ACK      - Transaction ID 0xdeed530d
```

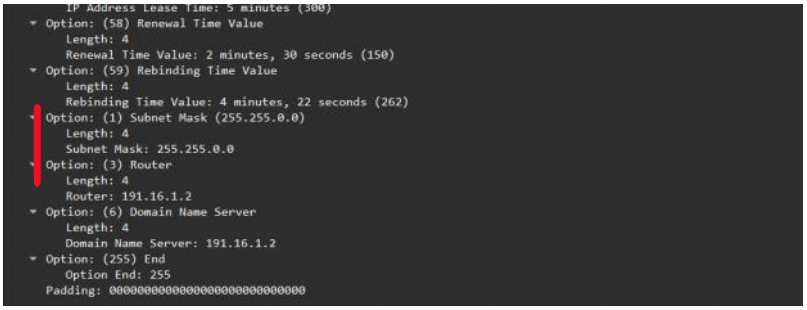
```
UDP payload (300 bytes)
Dynamic Host Configuration Protocol (Offer)
    Message type: Boot Reply (2)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0xdeed530d
    Seconds elapsed: 0
  ▸ Bootp flags: 0x0000 (Unicast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 191.16.11.129
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: Intel_0f:cd:e1 (2c:6d:c1:0f:cd:e1)
    Client hardware address padding: 00000000000000000000
    Server host name not given
    Boot file name not given
    Magic cookie: DHCP
  ▾ Option: (53) DHCP Message Type (Offer)
        Length: 1
        DHCP: Offer (2)
  ▾ Option: (54) DHCP Server Identifier (191.16.1.2)
        Length: 4
        DHCP Server Identifier: 191.16.1.2
  ▾ Option: (51) IP Address Lease Time
        Length: 4
        IP Address Lease Time: 5 minutes (300)
  ▾ Option: (58) Renewal Time Value
        Length: 4
        Renewal Time Value: 2 minutes, 30 seconds (150)
  ▾ Option: (59) Rebinding Time Value
```

9. In the example screenshot in this assignment, there is no relay agent between the host and the DHCP server. What values in the trace indicate the absence of a relay agent? Is there a relay agent in your experiment? If so what is the IP address of the agent?
   - A value of 0.0.0.0 indicates that there is no relay agent .

10. Explain the purpose of the router and subnet mask lines in the DHCP offer message.

```
302 14.301297    191.16.11.129    191.16.1.2         DHCP    342 DHCP Release  - Transaction ID 0x8e
344 20.077021    0.0.0.0          255.255.255.255    DHCP    345 DHCP Discover - Transaction ID 0xde
379 21.080930    191.16.1.2       191.16.11.129      DHCP    342 DHCP Offer    - Transaction ID 0xde
381 21.083443    0.0.0.0          255.255.255.255    DHCP    372 DHCP Request  - Transaction ID 0xde
382 21.091061    191.16.1.2       191.16.11.129      DHCP    342 DHCP ACK      - Transaction ID 0xde
```

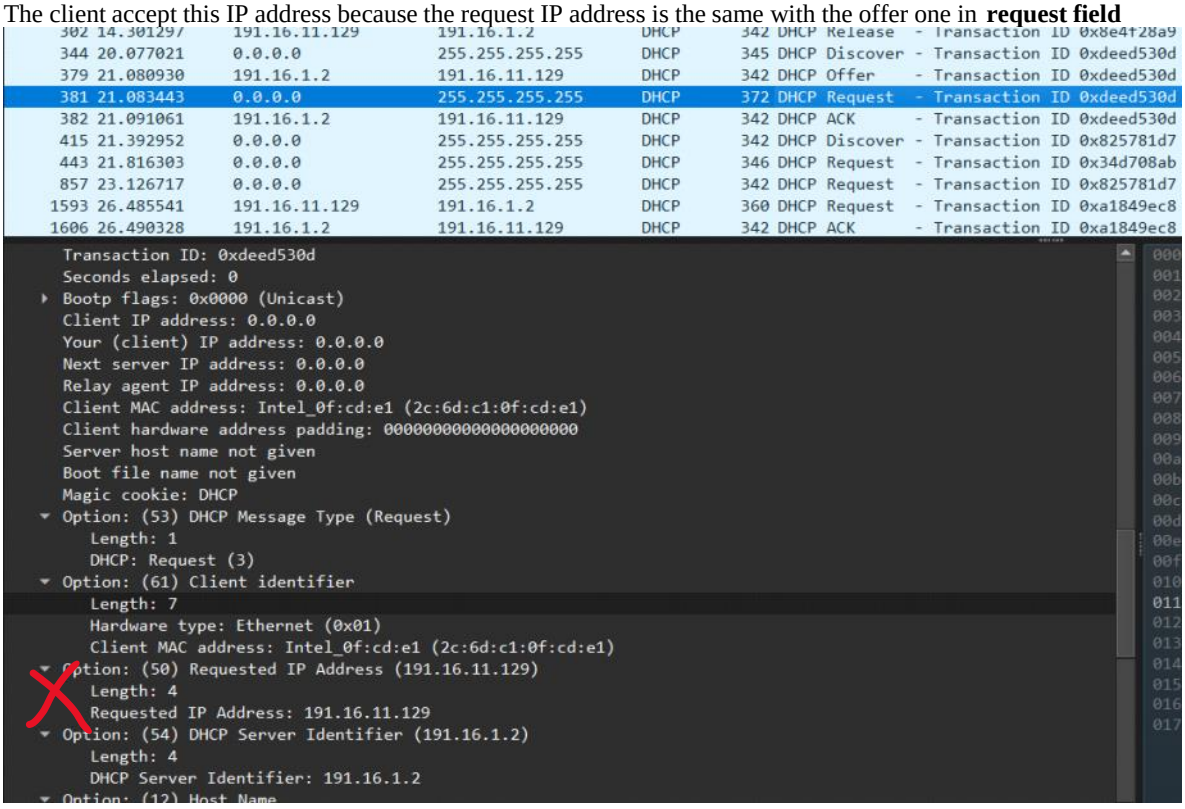
```
Server host name not given
Boot file name not given
Magic cookie: DHCP
▾ Option: (53) DHCP Message Type (Offer)
    Length: 1
    DHCP: Offer (2)
▾ Option: (54) DHCP Server Identifier (191.16.1.2)
    Length: 4
    DHCP Server Identifier: 191.16.1.2
▾ Option: (51) IP Address Lease Time
    Length: 4
    IP Address Lease Time: 5 minutes (300)
▾ Option: (58) Renewal Time Value
    Length: 4
    Renewal Time Value: 2 minutes, 30 seconds (150)
▾ Option: (59) Rebinding Time Value
    Length: 4
```



- The router line indicates where the client should send messages by default.
- The subnet mask line tells the client which subnet mask to use.

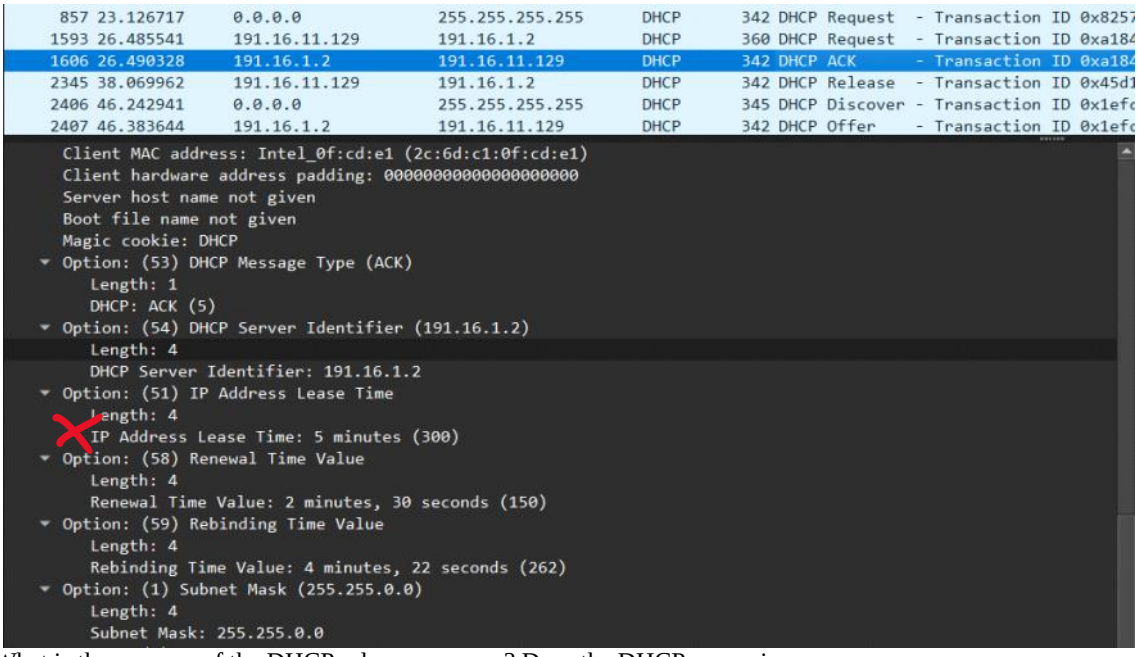
11. In the DHCP trace file noted in footnote 2, the DHCP server offers a specific IP address to the client (see also question 8. above). In the client's response to the first server OFFER message, does the client accept this IP address? Where in the client's RESPONSE is the client's requested address?

- The client accept this IP address because the request IP address is the same with the offer one in **request field**



12. Explain the purpose of the lease time. How long is the lease time in your experiment?

- A lease time is supplied along with the configuration information that DHCP transmits to a client. The client is permitted to use the allocated IP address for this amount of time
- The lease time in my experiment: 5 minutes

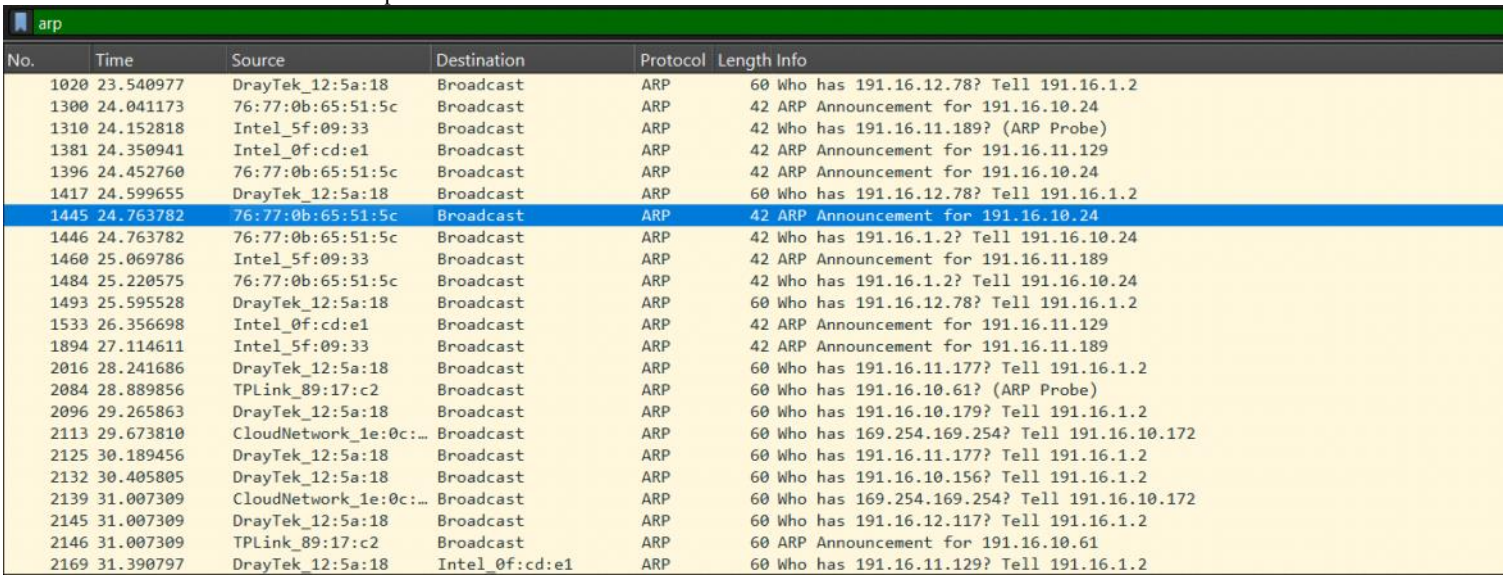


13. What is the purpose of the DHCP release message? Does the DHCP server issue an acknowledgment of receipt of the client's DHCP request? What would happen if the client's DHCP release message is lost?

- DHCP Release Message is the request to release the IP back to the DHCP Server.
- There is no ACK for this.
- Nothing happens if the release message is lost. The client will continue operation unitl its IP lease expires.

14. Clear the bootp filter from your Wireshark window. Were any ARP packets sent or received during the DHCP packet-exchange period? If so, explain the purpose of those ARP packets.

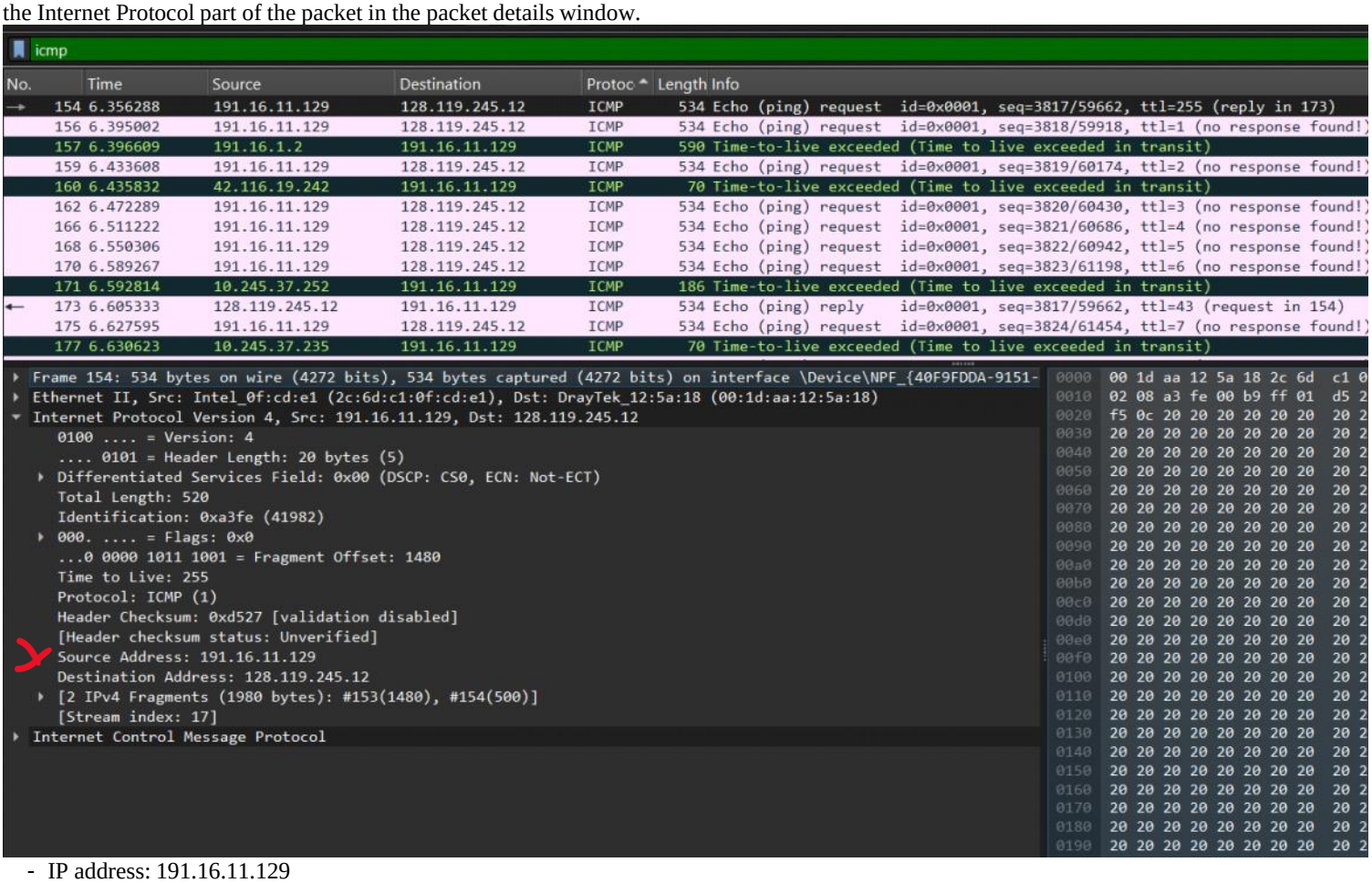
- Yes, there are many ARP packets
- An ARP request is sent when a device needs a MAC address associated with an IP address, and it does not have an entry for the IP address in its ARP table. This is used to map MACs to IPs in the local network.



Lab 5a - IP

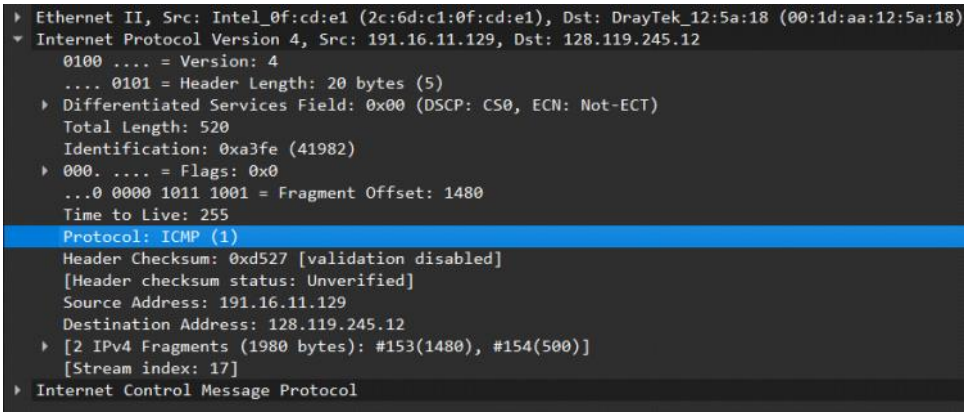
Thursday, October 24, 2024 5:00 PM

1. Select the first ICMP Echo Request message sent by your computer, and expand the Internet Protocol part of the packet in the packet details window.

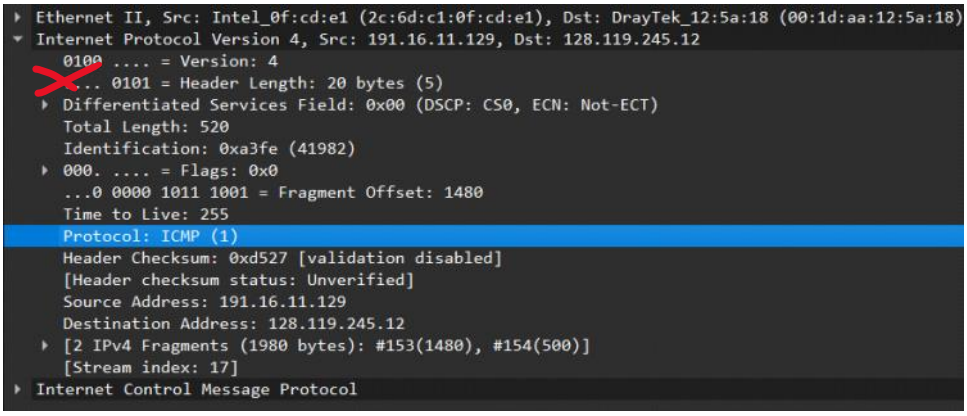


- IP address: 191.16.11.129

2. Within the IP packet header, what is the value in the upper layer protocol field?
   - The value of the upper layer protocol field is ICMP (0X01)



3. How many bytes are in the IP header? How many bytes are in the payload of the IP datagram? Explain how you determined the number of payload bytes.
   - There are 20 bytes in the IP header which leaves 36 bytes for the payload of the IP datagram because we were sending a packet of length 56 bytes.



4. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented.

- The fragment offset is set to 0, therefore, the packet has not been fragmented.

5. Which fields in the IP datagram always change from one datagram to the next within this series of ICMP messages sent by your computer?
It is the header checksum and identification

6. Which fields stay constant? Which of the fields must stay constant? Which fields must change? Why?
Constant:

- Version (IPv4)
- Header length
- IP source (because of sending from same place)
- IP destination (because of contacting the same site_
- Upper layer protocol (always using ICMP)

Must stay constant:

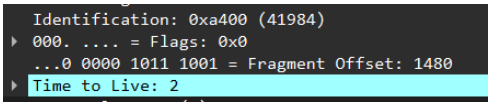
- Same as above

Must change:

- Header checksum (header changes)
- Identification (to verify packets)

7. Describe the pattern you see in the values in the Identification field of the IP datagram
The identification field's value increases by 1 in each request

8. What is the value in the Identification field and the TTL field?

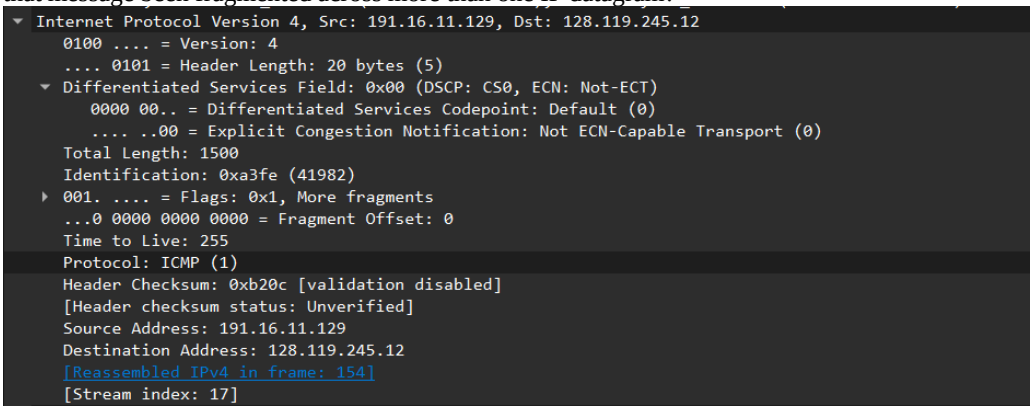


- Identification: 41984
- TTL: 2

9. Do these values remain unchanged for all of the ICMP TTL-exceeded replies sent to your computer by the nearest (first hop) router? Why?

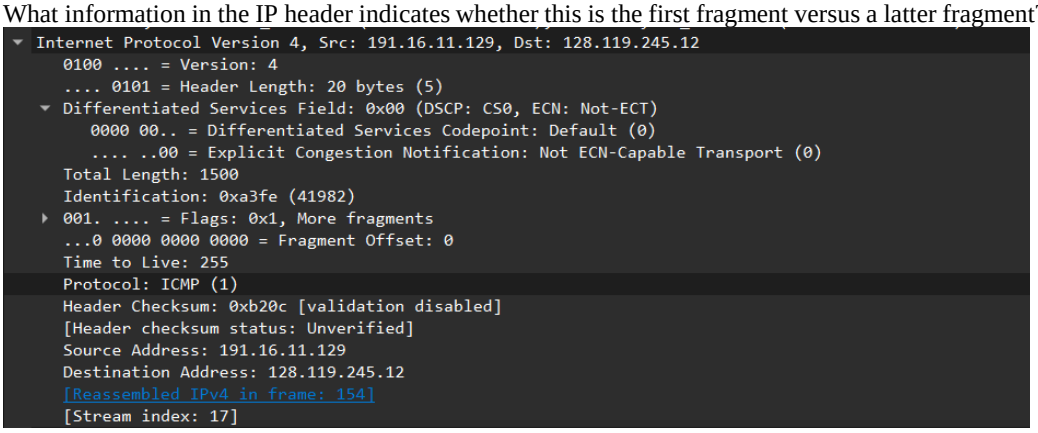
- The Identification field varies across all replies since it needs to have a unique value. If two or more replies share the same value, it indicates that they are fragments of a larger packet.
- The TTL field remains unchanged because the time to live for the first hop router is consistent.

10. Find the first ICMP Echo Request message that was sent by your computer after you changed the Packet Size in pingplotterto be 2000. Has that message been fragmented across more than one IP datagram?



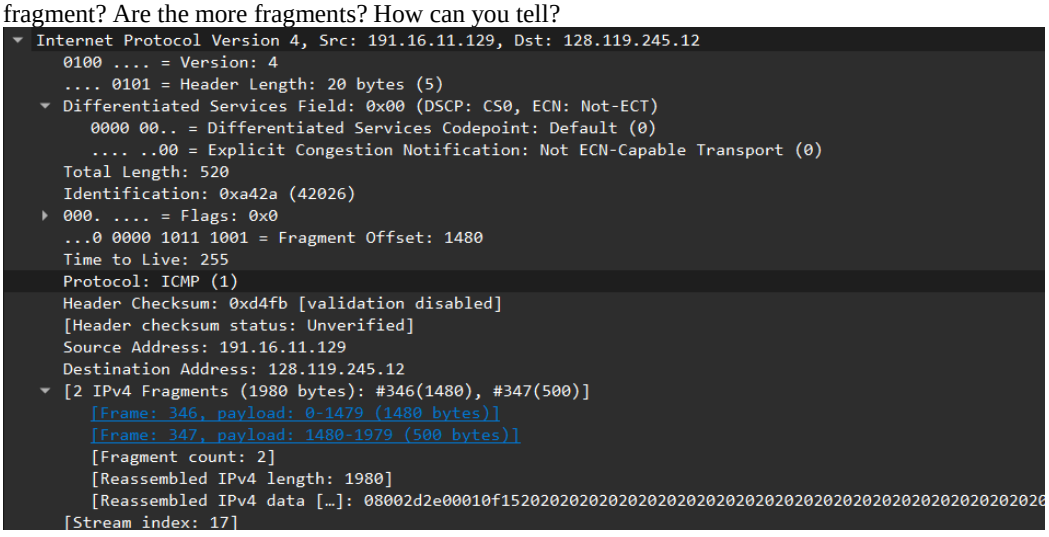
Yes, the message is fragmented into more than 1 diagram

11. Print out the first fragment of the fragmented IP datagram. What information in the IP header indicates that the datagram been fragmented? What information in the IP header indicates whether this is the first fragment versus a latter fragment? How long is this IP datagram?



The presence of the "more fragments" flag indicates that the datagram has been fragmented. The fragment offset is set to 0, meaning this is the initial fragment, unlike subsequent fragments where this value would be set to 1480. The total length of the datagram is 1500.

12. Print out the second fragment of the fragmented IP datagram. What information in the IP header indicates that this is not the first datagram fragment? Are the more fragments? How can you tell?



The second fragment is identifiable by its fragment offset of 1480. There are no additional fragments, as the "more fragments" flag is no longer set.

13. What fields change in the IP header between the first and second fragment?

- Length
- Flags set
- Fragment offset
- Header checksum

14. How many fragments were created from the original datagram?

- 3 fragments

15. What fields change in the IP header among the fragments?

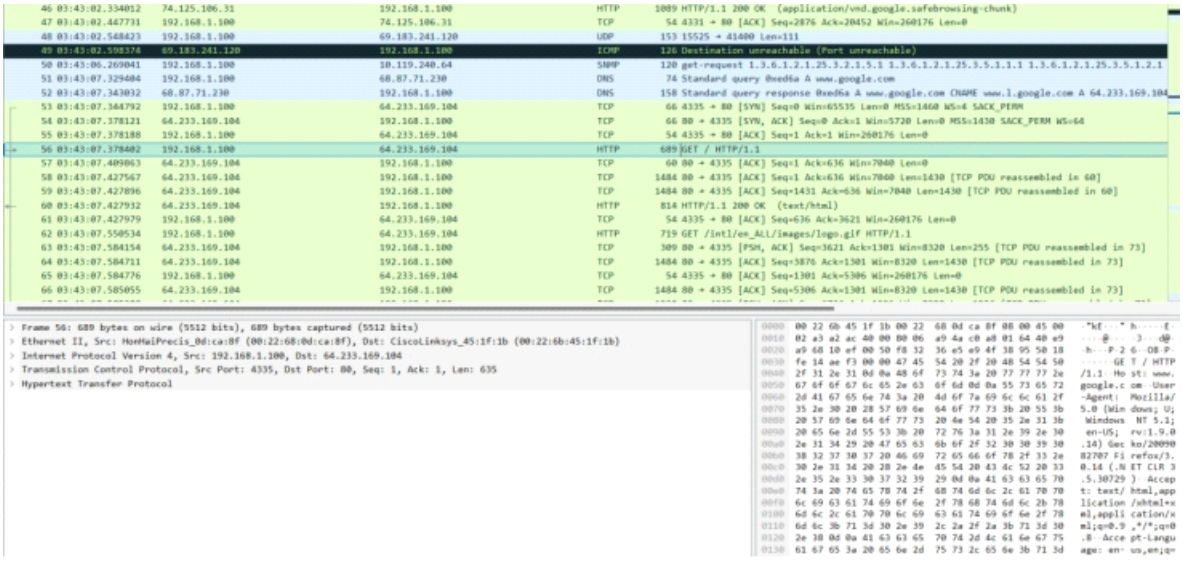
- The second fragment is identifiable by its fragment offset of 1480. There are no additional fragments, as the "more fragments " flag is no longer set.

Lab 5c - NAT

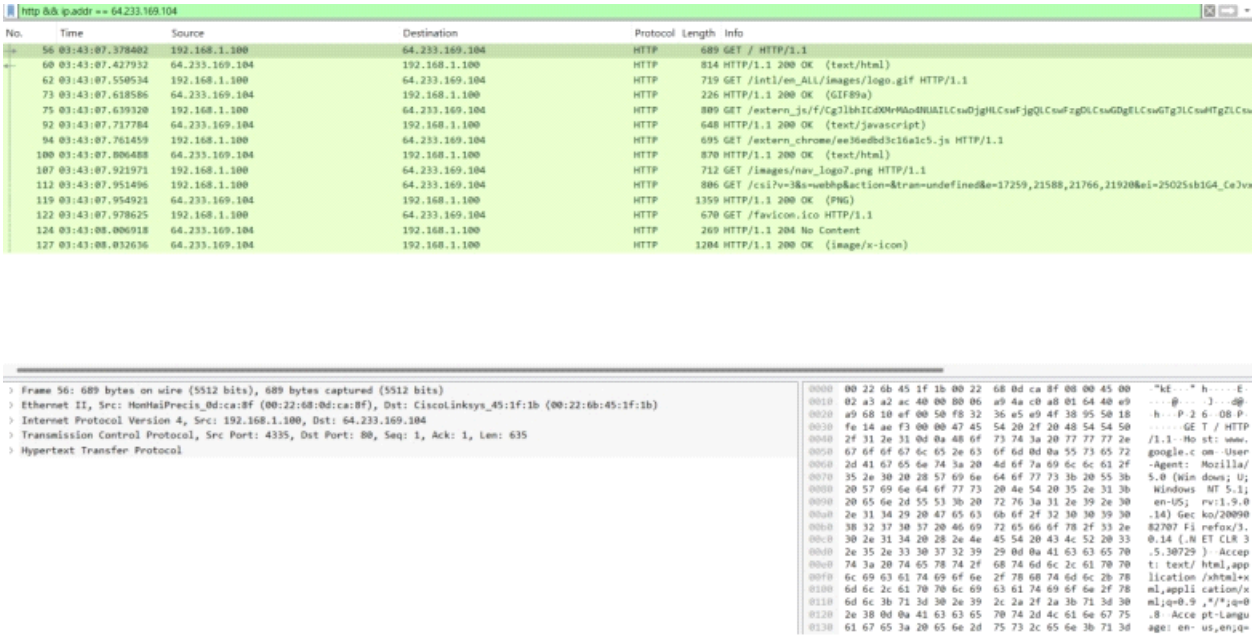
Thursday, October 24, 2024 5:50 PM

What is the IP address of the client?

IP address of client: 192.168.1.100



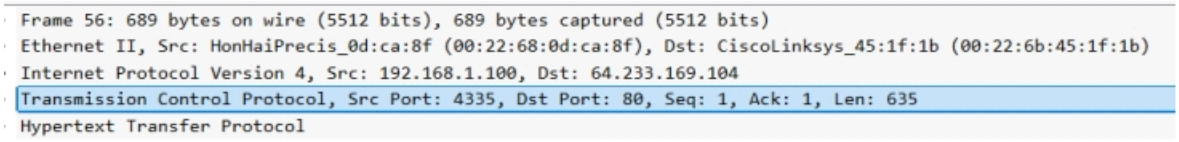
The client actually communicates with several different Google servers in order to implement “safe browsing.” (See extra credit section at the end of this lab). The main Google server that will serve up the main Google web page has IP address 64.233.169.104. In order to display only those frames containing HTTP messages that are sent to/from this Google, server, enter the expression “http && ip.addr == 64.233.169.104” (without quotes) into the Filter: field in Wireshark .



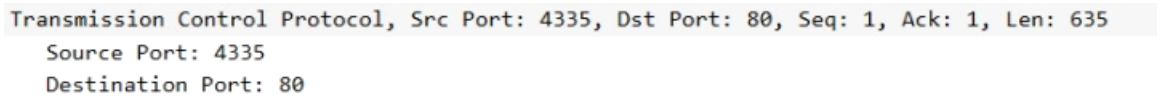
Consider now the HTTP GET sent from the client to the Google server (whose IP address is IP address 64.233.169.104) at time 7.109267. What are the source and destination IP addresses and

TCP source and destination ports on the IP datagram carrying this HTTP GET?

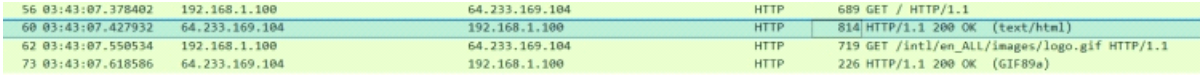
Source IP address: 192.168.1.100, destination IP address: 64.233.169.104

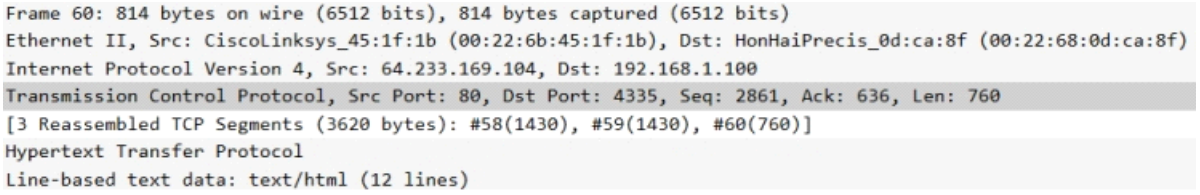


TCP source port: 4335 , TCP destination port: 80



At what time is the corresponding 200 OK HTTP message received from the Google server? What are the source and destination IP addresses and TCP source and destination ports on the IP datagram carrying this HTTP 200 OK message?





Time: 03:43:07.427932

Source IP: 64.233.169.104, Destination IP: 192.168.1.100

Source port: 80, Destination port: 4335

Recall that before a GET command can be sent to an HTTP server, TCP must first set up a connection using the three-way SYN/ACK handshake. At what time is the client-to-server TCP SYN segment sent that sets up the connection used by the GET sent at time 7.109267? What are the source and destination IP addresses and source and destination ports for the TCP SYN segment? What are the source and destination IP addresses and source and destination ports of the ACK sent in response to the SYN. At what time is this ACK received at the client? (Note: to find these segments you will need to clear the Filter expression you entered above in step 2. If you enter the filter “tcp”, only TCP segments will be displayed by Wireshark).

SYN segment:

Source IP address: 192.168.1.100

Destination IP address: 64.233.169.104

Source port: 4335

Destination port: 80

Time: 03:43:07.244792

ACK response:

Source IP address: 64.233.169.104

Destination IP address: 192.168.1.100

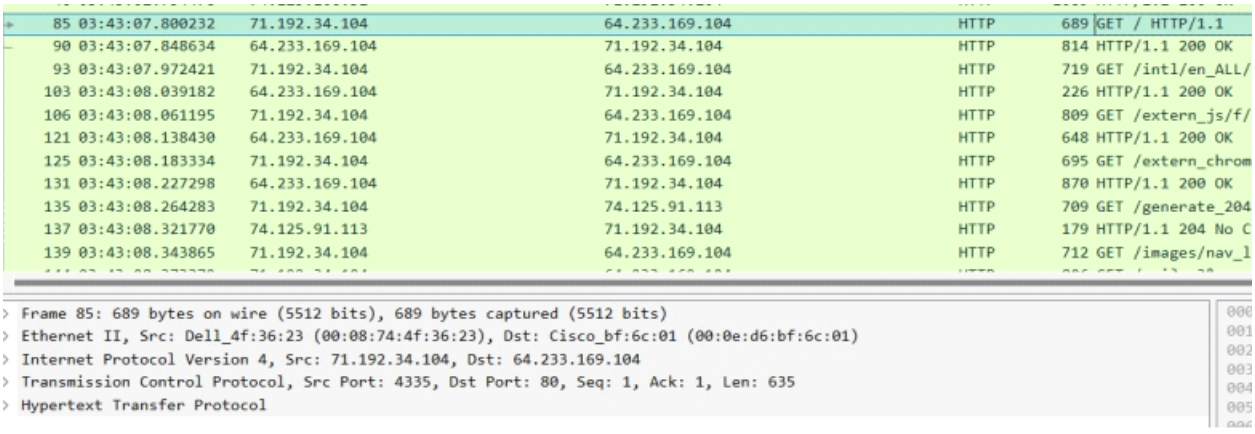
Source port: 80

Destination port: 4335

Time: 03:43:07.378121

ACK received at the client at time: 03:43:07.378188

In the NAT_ISP_side trace file, find the HTTP GET message was sent from the client to the Google server at time 7.109267 (where t=7.109267 is time at which this was sent as recorded in the NAT_home_side trace file). At what time does this message appear in the NAT_ISP_side trace file? What are the source and destination IP addresses and TCP source and destination ports on the IP datagram carrying this HTTP GET (as recording in the NAT_ISP_side trace file)? Which of these fields are the same, and which are different, than in your answer to question 3 above?



Time sent: 03:43:07.800232

Source IP: 71.192.34.104, Destination IP: 64.233.169.104

Source port: 4335, Destination port: 80

The IP address of source has changed compared to question 3.

Are any fields in the HTTP GET message changed? Which of the following fields in the IP datagram carrying the HTTP GET are changed: Version, Header Length, Flags, Checksum. If any of these fields have changed, give a reason (in one sentence) stating why this field needed to change.

No fields in the HTTP GET message has changed

The checksum field in the IP datagram carrying the HTTP GET is changed

The checksum change because the IP source address has changed

In the NAT_ISP_side trace file, at what time is the first 200 OK HTTP message received from

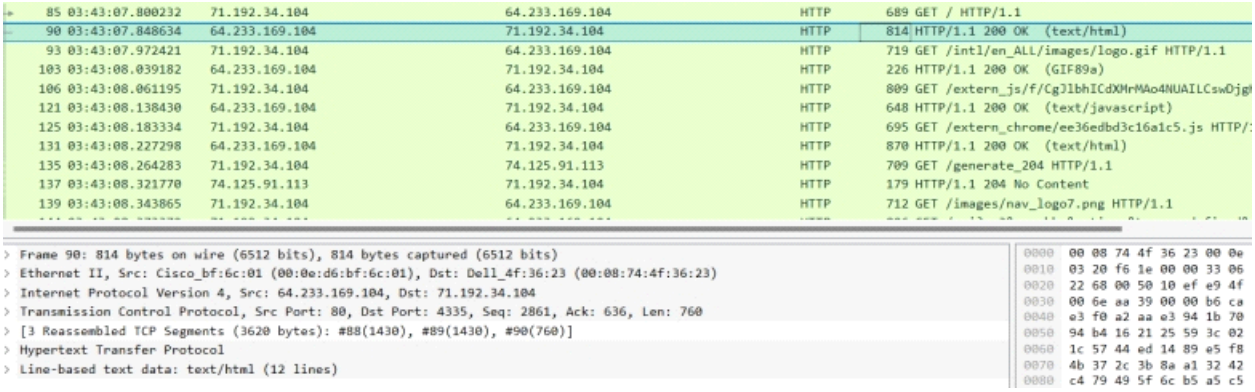
the Google server? What are the source and destination IP addresses and TCP source and destination ports on the IP datagram carrying this HTTP 200 OK message? Which of these fields are the same, and which are different than your answer to question 4 above?

Time: 03:43:07.848634

Source IP address: 64.233.169.104, Destination IP address: 71.192.34.104

Source port: 80, Destination port: 4335

The destination IP address has changed



In the NAT_ISP_side trace file, at what time were the client-to-server TCP SYN segment and the server-to-client TCP ACK segment corresponding to the segments in question 5 above captured? What are the source and destination IP addresses and source and destination ports for these two segments? Which of these fields are the same, and which are different than your answer to question 5 above?

SYN segment

Time: 03:43:07.766539

Source IP: 71.192.34.104, Destination IP: 64.233.169.104

Source port: 4335, Destination port: 80

ACK segment

Time: 03:43:07.798839

Source IP: 64.233.169.104, Destination IP: 71.192.34.104

Source port: 80 , Destination port: 4335

For the SYN, the source IP address has changed, while for the ACK, the destination IP address has changed

Using your answers to 1-8 above, fill in the NAT translation table entries for HTTP connection considered in questions 1-8 above.

NAT translation table

WAN: 71.192.34.104, 4335

LAN: 192.168.1.100, 4335