School of Electrical and Electronic of Enginnering
EEE 4474-01 : Experiments on Communication Networks
2조 2016142096 조윤신, 2017142043 김재민

Result Report Week 2

wireshark(2) : TCP UDP IP protocols

Experiment 1 : TCP

Topic 1-1 : A first look at the captured trace

We used wireshark's given captured packet file 'TCP-ethreal-trace-1' for experiment 1 answering problem 1 to 12.

Problems

Problem 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"

Answer Source IP address : 192. 168.1.102 / Souce port : 1161

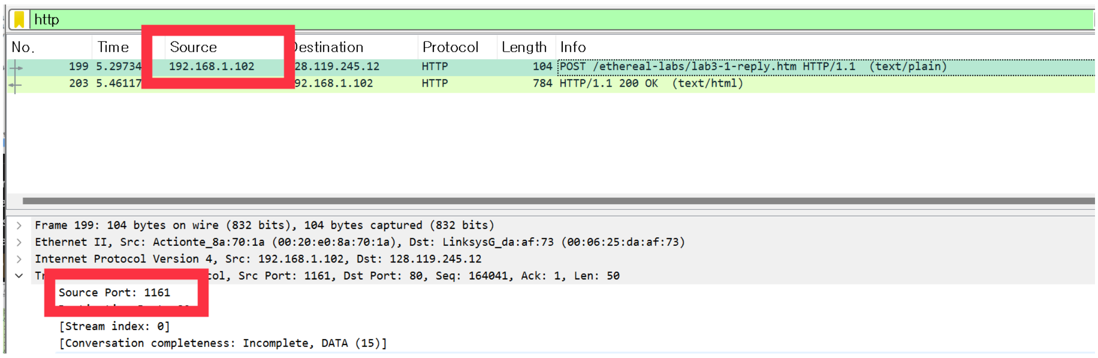


Figure 1: Problem 1-1's screenshot : Packet - POST / reply (text/plain)

Problem 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?

Answer Destination IP address : 128. 119.245.12 / Destination port : 80

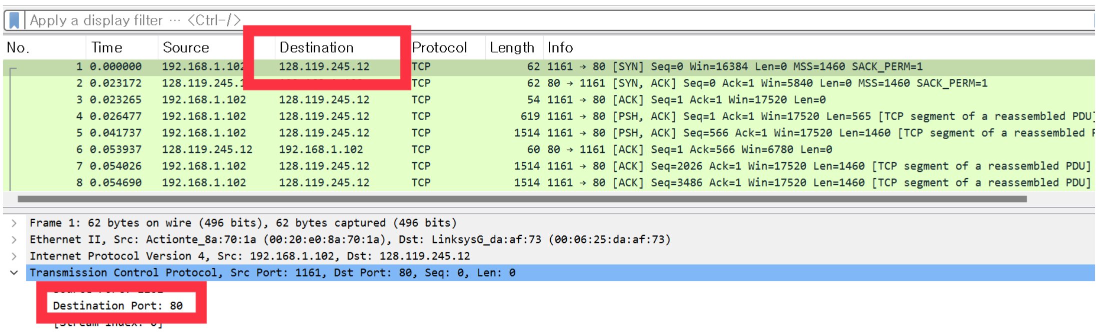


Figure 2: Problem 1-2's screenshot : Packet - [SYN] Seq = 0

Topic 1-2 : TCP Basics

Problems

Problem 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?

Answer The sequence number of TCP SYN segment that is used to initate the TCP connection of that is the no.1 Segement in filtered packet list by keyword, 'TCP' is the value of 0.
We can figure out that segment is a SYN segment as that of TCP header contains Flags value.

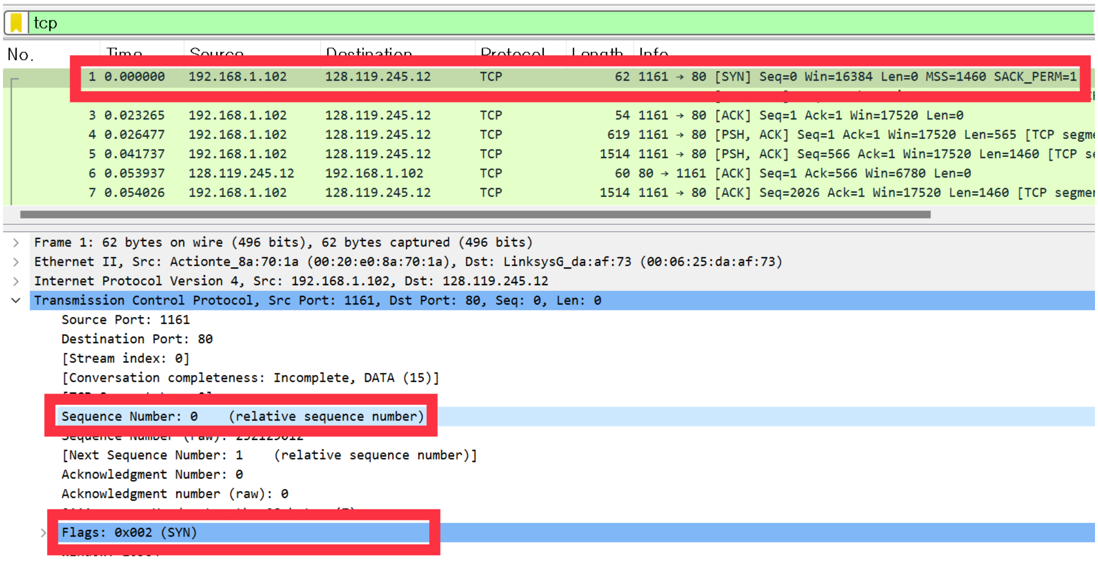


Figure 3: Problem 1-3's screenshot : Packet - [SYN] seq = 0's TCP header

Problem 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 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 is 0. Acknowledgement field is the value of sequence number plus 1, 1.
The message contains the information of this segment is the SYN,ACK segment as marked figure below.

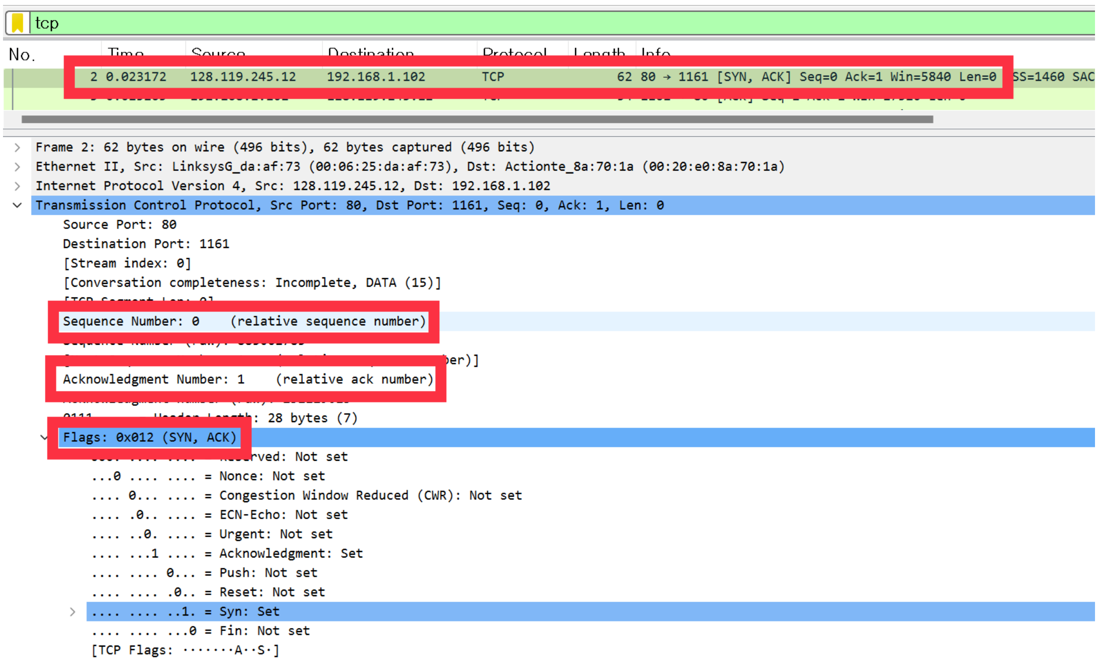


Figure 4: Problem 1-4's screenshot : Packet - HTTP POST's TCP Header

Problem 5: 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.

Answer The sequence number of the TCP segment containing the HTTP POST command is 164041.

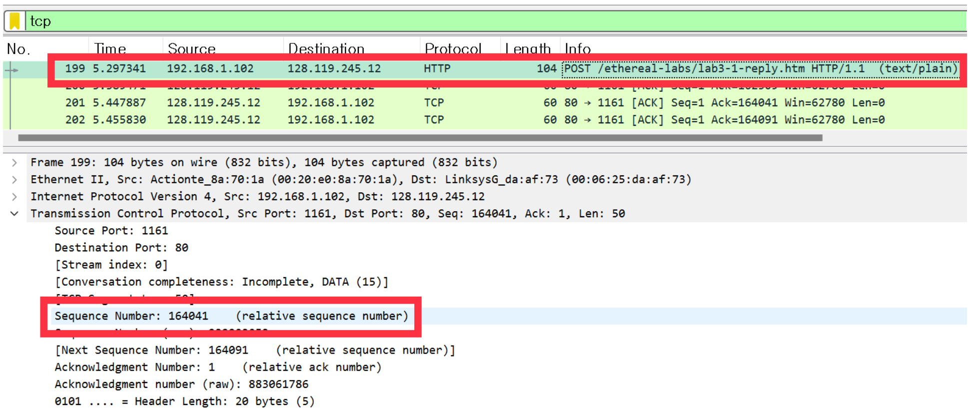


Figure 5: Problem 1-5's screenshot : Packet - HTTP POST's TCP Header

Problem 6: 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 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 below for all subsequent segments.

$$\text{Estimated RTT} = 0.875 \times \text{Estimated RTT} + 0.125 \times \text{Sample RTT}$$

Answer The first six segements are No. 4,5,7,8,10,11. And those of sequence number are 1, 566, 2026, 3486, 4946, 6406.

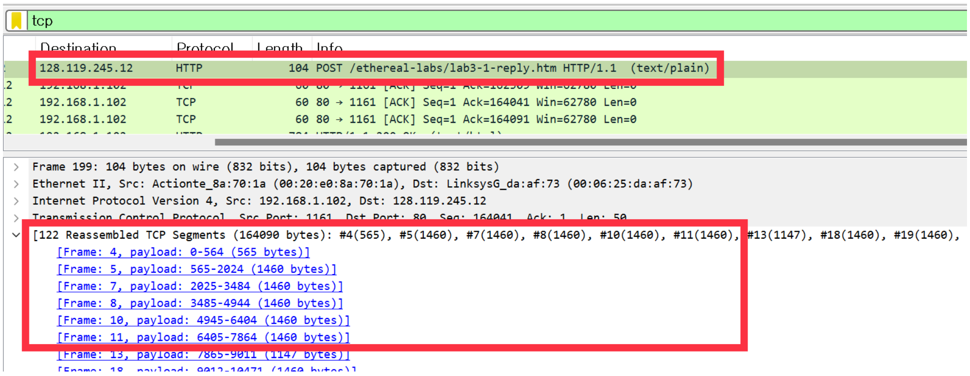


Figure 6: Problem 1-6-1's screenshot : Packet - HTTP POST's TCP Header with 122 of reassembled segments's sequence

The time of segment sent, segment received the ACK, and the value of RTT are taken table below. To know each of six segment's ACK received time, the No. of ACKs that segemnts received are No. 6, 9,12,14,15,16.

	Sent Time	Ack Received Time	RTT (ACK Received TIme - Sent Time)
Segment 1	0.026477	0.053937	0.027460
Segment 2	0.041737	0.077294	0.035557
Segment 3	0.054026	0.124085	0.070059
Segment 4	0.054690	0.169118	0.114430
Segment 5	0.077405	0.217299	0.139890
Segment 6	0.078157	0.267802	0.189640

Table 1: The calculated value of RTT with the first six segments

The estimated RTT is calculated by given equation.

Problem 6

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 below for all subsequent segments.

$$\text{EstimatedRTT} = 0.875 \times \text{EstimatedRTT} + 0.125 \times \text{SampleRTT}$$

```python
# The value of RTT for the first six segments.
RTT = [0.027460, 0.035557, 0.070059, 0.114430, 0.139890, 0.189640]
# Given EstimatedRTT function
func_EstimatedRTT = lambda x_1,x_2: 0.875*x_1 + 0.125*x_2

EstimatedRTT = [RTT[0]]
for i in range(len(RTT[1:])):
    EstimatedRTT.append(round(func_EstimatedRTT(EstimatedRTT[i],RTT[i+1]),5))
EstimatedRTT
```

[1] ✓ 0.1s Python

```
[0.02746, 0.02847, 0.03367, 0.04376, 0.05578, 0.07251]
```

```python
> for i in range(len(RTT)):···
```

[2] ✓ 0.1s Python

```
EstimatedRTT after the receipt of the ACK of segment 1
    EstimatedRTT  =  0.02746 (sec)
EstimatedRTT after the receipt of the ACK of segment 2
    EstimatedRTT  =  0.02847 (sec)
EstimatedRTT after the receipt of the ACK of segment 3
    EstimatedRTT  =  0.03367 (sec)
EstimatedRTT after the receipt of the ACK of segment 4
    EstimatedRTT  =  0.04376 (sec)
EstimatedRTT after the receipt of the ACK of segment 5
    EstimatedRTT  =  0.05578 (sec)
EstimatedRTT after the receipt of the ACK of segment 6
    EstimatedRTT  =  0.07251 (sec)
```

Figure 7: Problem 1-6-2's screenshot : The calculation result of EstimatedRTT by jupyter notebook

We plot the RTT for each of the TCP segments that were being sent from the client to the gaia.cs.umass.edu.server.

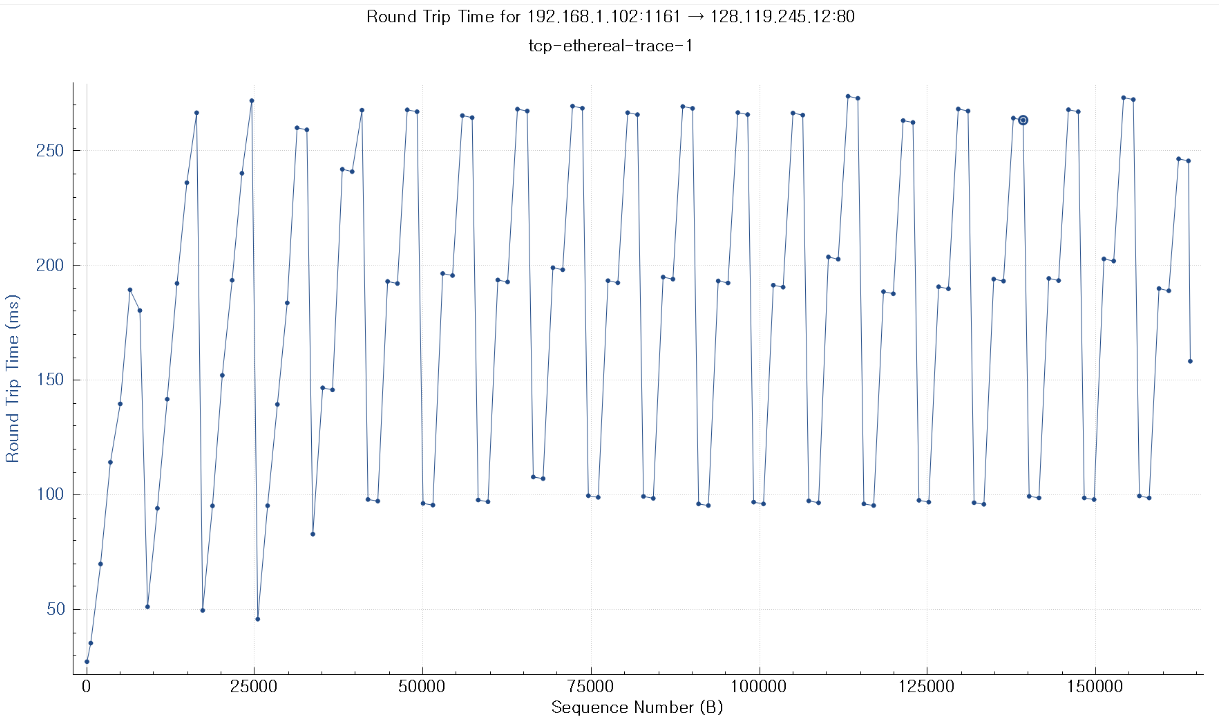


Figure 8: Problem 1-6-3's screenshot : RTT plot

Problem 7: What is the length of each of the first six TCP segments?

Answer The length [1] of the fitst TCP segments is 565, and the other TCP segments are 1460 as same.

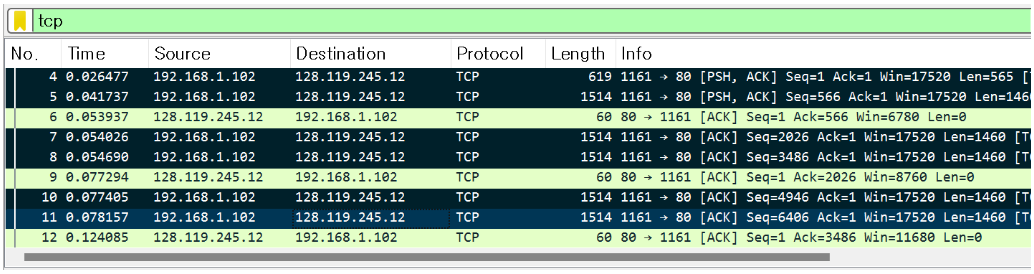


Figure 9: Problem 1-7's screenshot : Packet List - Marked the first six TCP segments, No.4, 5, 7, 8, 10, 11

Problem 8: 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?

Answer The minimum amount of available buffer space advertised at the received for the entire trace be marked by the first ACK sent from the server. And the value is the window size value of the ACK. The first ACK, No.6 packet, of value is 6780.

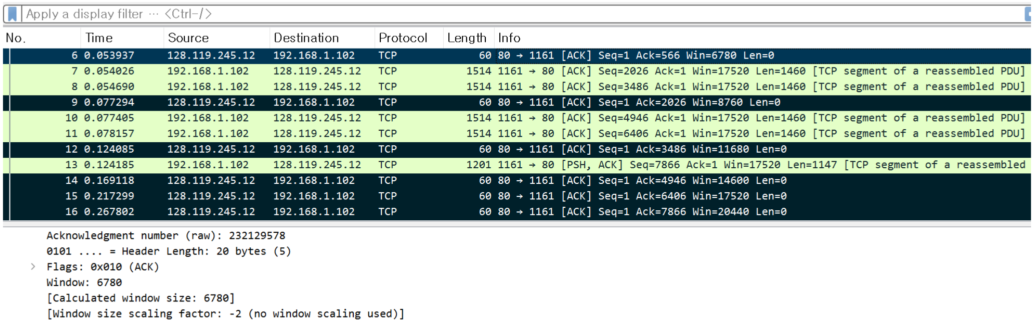


Figure 10: Problem 1-8's screenshot : Packet List - Marked the first six ACK, No.6, 9, 12, 14, 15, 16 with ACK No.6's message

Since we can find out that the first six ACK's window size grows up to 20440 at ACK No.16 , and that means the maximum had not been reached in given trace. There was no throrrled because of the lack of receiver buffer space.

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

Answer From the Sequnce-Time Plot we can figure out that the Sequnce number arrived enumerating by time, just steady increase so that there were no retransmitted segments.

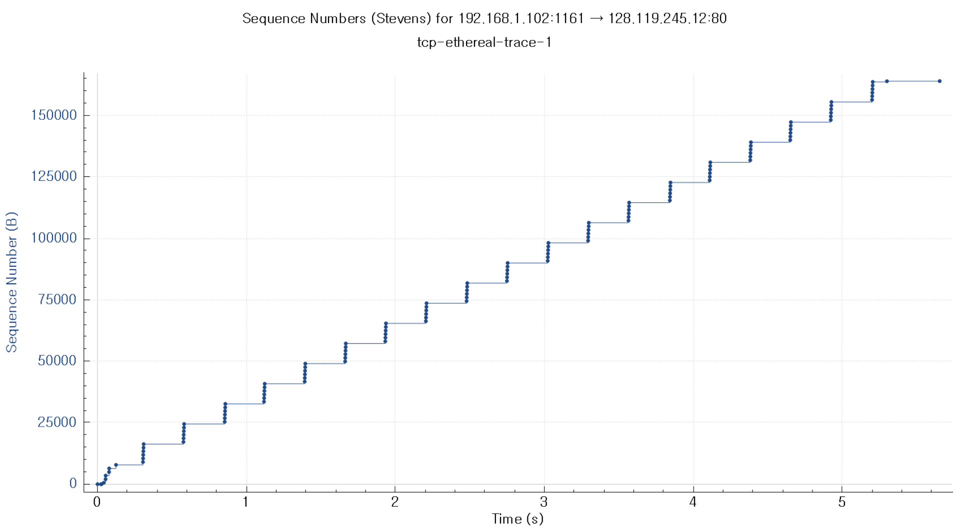


Figure 11: Problem 1-9's screenshot :

[1] 'Len' in packet info in Figure 9

Problem 10: How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment.

Answer As we know that the ACK number is that the client puts is the sequence number of the next byte the client expecting from the receiver. That it is the difference between the continous ACK is the amount of the data.

	ACK number	acknowledged data (bytes)
ACK 1	566	566
ACK 2	2026	1460
ACK 3	3486	1460
ACK 4	4946	1460
ACK 5	6406	1460
ACK 6	7866	1460

Table 2: The first six ACKS and their acknowledged sequenxw number & acknowleded data

Problem 11: What is the throughput for the TCP connection? Explain how you calculated this value.

Answer The Troughput can be calculated by the equation below:

$$\text{Throughput} = \frac{\text{Amount of data transmitted}}{\text{Time incurred}}$$

The amount of the HTTP data ckient sent, the last ACK's packet 202's ACK number,164091, the value of the opposite expectating data sequence number. [2].The incurred can be calculated by the time difference between the first deassembled ACK, Packet No. 4, and the last ACK, Packet No.202, Time of packet No.202– Time of packet No.4 = 5.455830 – 0.026477 = 5.429353(sec)

$$\text{Throughput} = \frac{[\text{Amount of data transmitted}] = 164091 \text{ (bytes)}}{[\text{Time incurred}] = 5.429353(sec)} = 30,222.938 \text{ Bytes}/sec$$

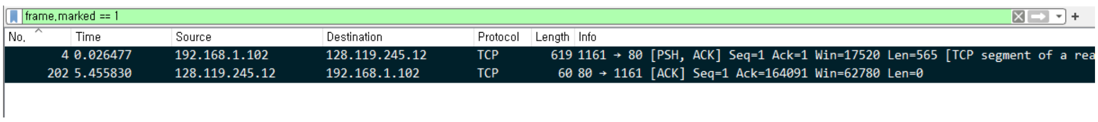


Figure 12: Problem 1-11's screenshot : No.4 and No.202 packet's

[2]Amount of the data transmitted in problem

Topic 1-3 : TCP congestion control in action

Problems

Problem 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. Can you identify where TCP's slow-start 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.

Answer TCP slowstart phase begins on 0.026s and ends on 0.124s. After the slowstart phase, sequence number is steadily increasing without any congestion avoidance. In fact, the server is sending 6 packets at once. I guess that the HTTP server has the throughput restriction.

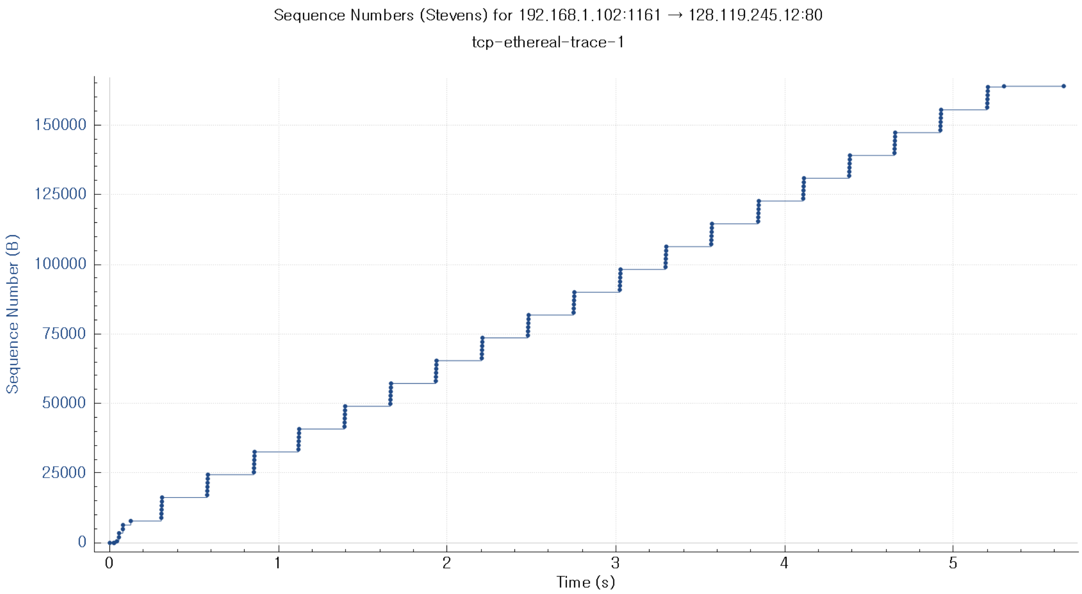


Figure 13: Problem 1-12's screenshot :

Experiment 2 : UDP

Topic 2-1 : The Assignment

Problems

Problem 1: Select one UDP packet from you r trace . From this packet, determine how many fields there are in the UDP header.

Answer There are 4 fields. : Source Port, Destination Port, Length, and Checksum

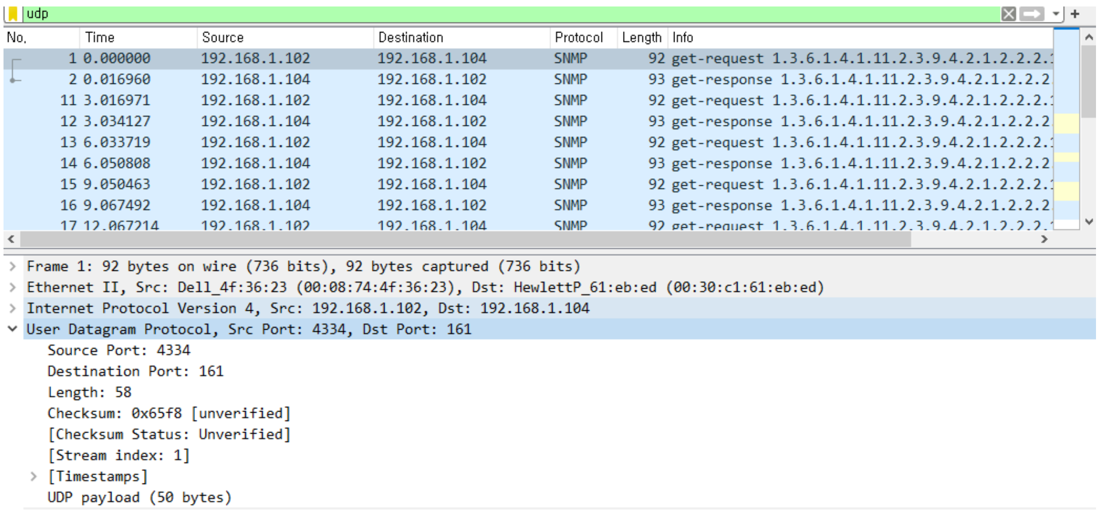


Figure 14: Problem 2-1's screenshot :

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

Answer The header length of UDP is that Length – UDP payload : 58 – 50 = 8.
As we can see in figure the 4 fields of header has the same length. Therefore each of 4 header fields is 2 bytes long.

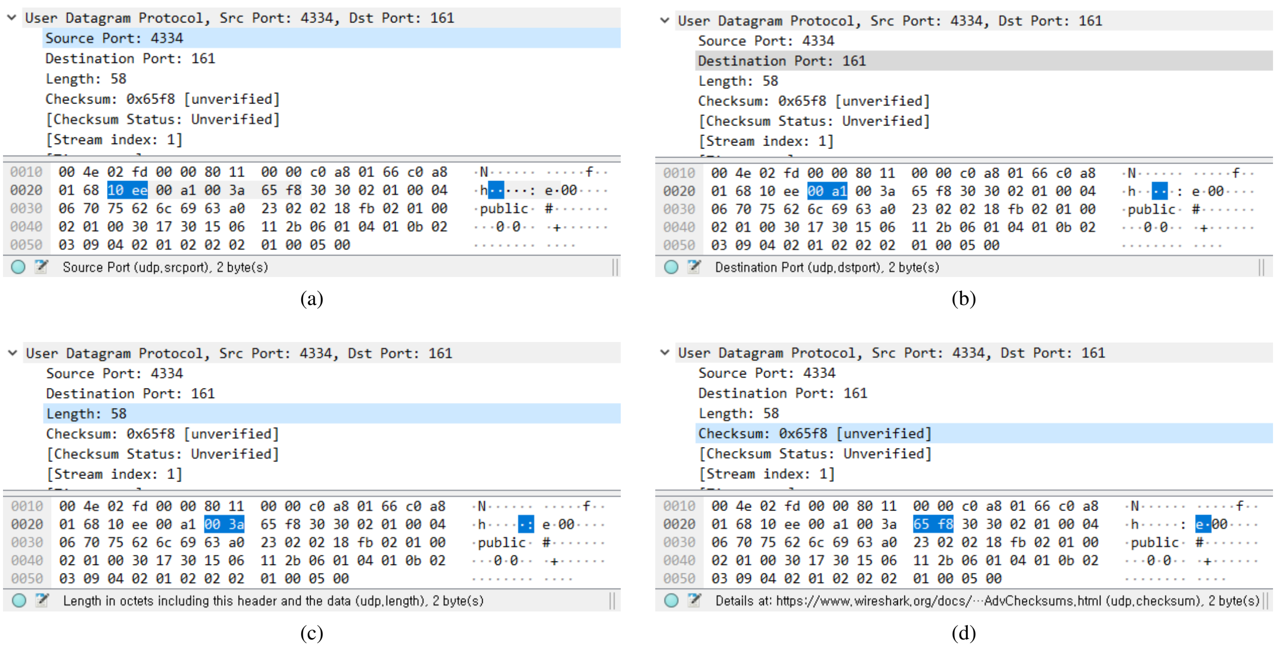


Figure 15: Problem 2-2's screenshot :

Problem 3: The value in the Length field is the length of what? this answer

Answer Length' is the length of the UDP header plus the UDP data. We can verify this from the packet below. Total length = header + data = 8 + 50 = 58(bytes)

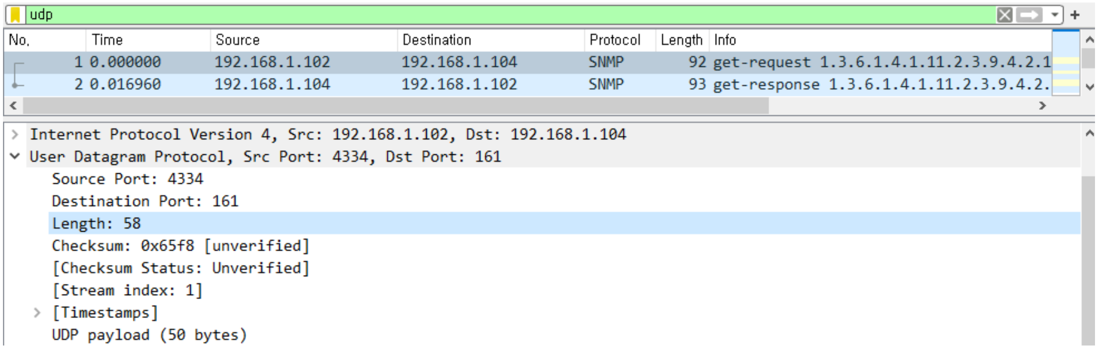


Figure 16: Problem 2-3's screenshot :

Problem 4: What is the maximum number of bytes that c

Answer UDP header's length field is 2 bytes (16 bit) long, so UDP's maximum length is $2^{16}-1 = 65535$ bytes. Since UDP header is 8 bytes long, UDP payload's maximum length is $65535 - 8 = 65537$ bytes .

Problem 5: What is the largest possible source port number?

Answer UDP header's source port field is 2 bytes (16 bit) long, so the largest possible source port number is $2^{16} - 1 = 65535$.

Problem 6: What is the protocol number for UDP? Give your answer in both hex decimal notation. To answer this question, you'll need to loo adecimal and k into the field of the IP datagram containing this UDP segment.

Answer hexadecimal notaion : 11 / decimal notation : 17

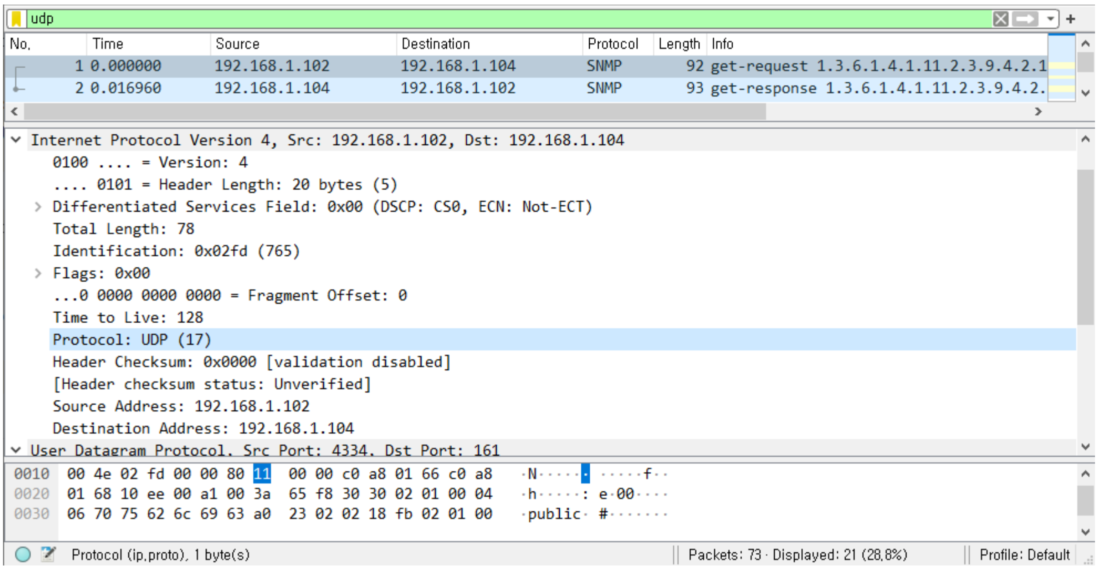


Figure 17: Problem 2-6's screenshot :

Experiment 3 : IP

Topic 3-1 : Capturing packets from an execution of traceroute

Problems

Problem 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. What is the IP address of your computer?
Answer IP address : 192.168.86.61

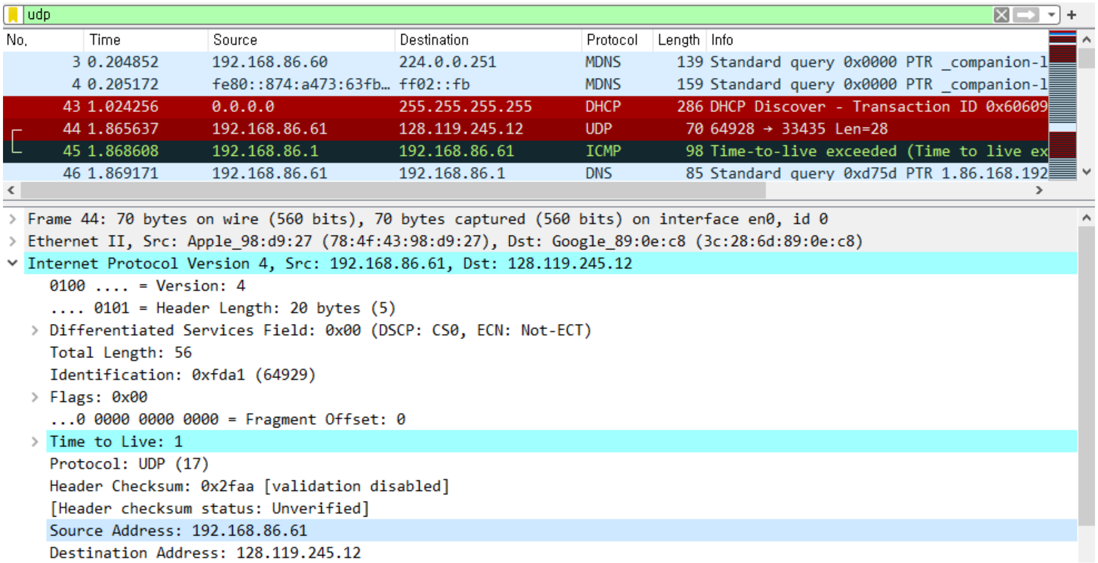


Figure 18: Problem 3-1's screenshot :

Problem 2: Within the IP packet header, what is the value in the upper layer protocol field?
Answer Protocol : UDP (17)

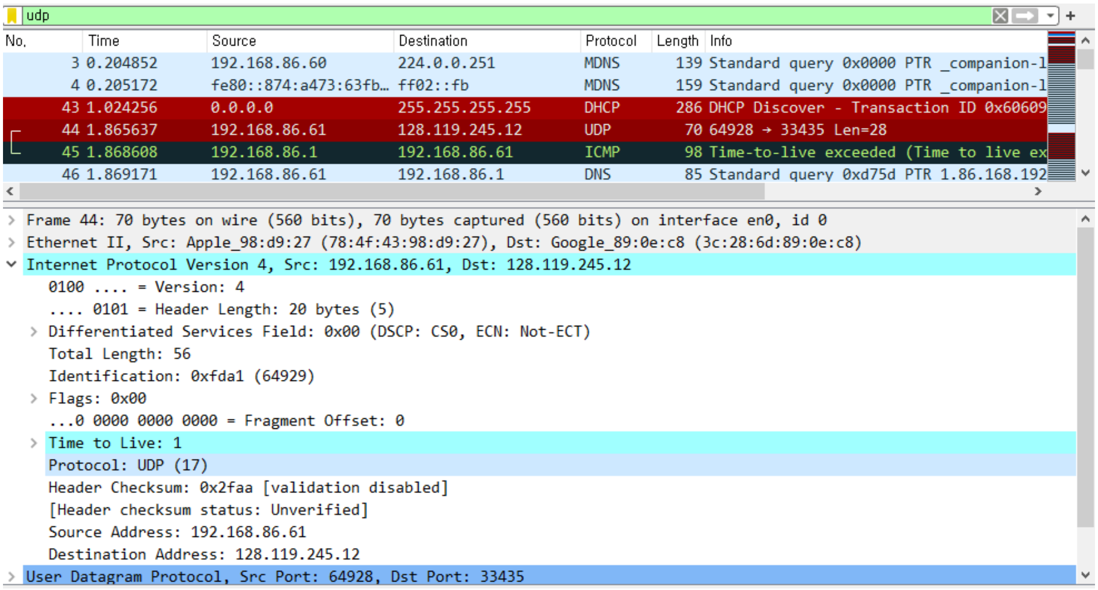


Figure 19: Problem 3-2's screenshot :

Problem 3: How many bytes are in the IP header?

Answer 20 bytes

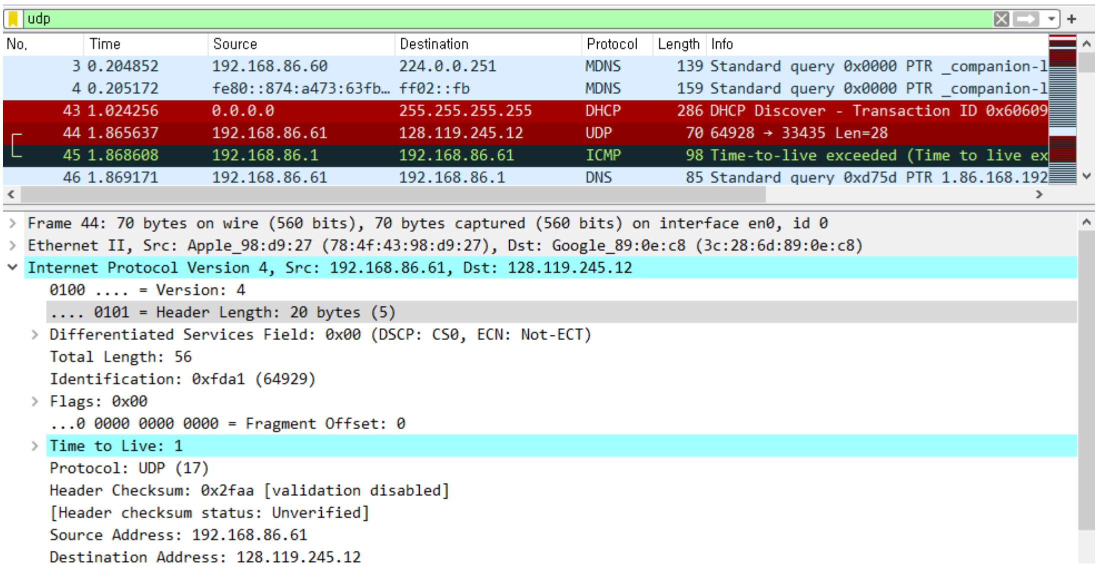


Figure 20: Problem 3-3's screenshot :

Problem 4: How many bytes are in the payload of the IP datagram? Explain how you determined the number of payload bytes.

Answer Since total length is 56 bytes long and the header is 20 bytes long, the length of payload is $56 - 20 = 36$ bytes

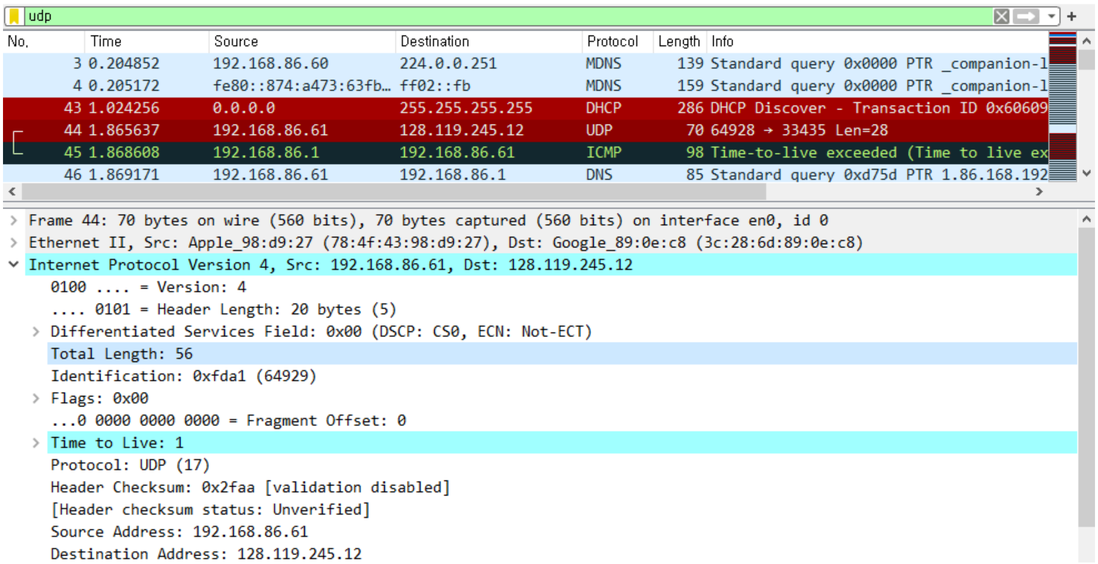


Figure 21: Problem 3-4's screenshot :

Problem 5: Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented.

Answer Since Flags More fragments bit = 0, this IP datagram hasn't been fragmented.

Topic 3-2 : Basic IPv4

Problem 6: Which fields in the IP datagram always change from one datagram to the next within this series of UDP segments sent by your computer destined to 128.119.245.12, via traceroute? Why?
Answer Identification : Each IP datagram has different identification number, unless it is fragmented.
Header Checksum : Checksum changes as the header changes.
Time to Live : ‘traceroute’ increases TTL with each subsequent packet.

Problem 7: Which fields in this sequence of IP datagrams (containing UDP segments) stay constant? Why?
Answer Version : They are all IPv4 datagrams.
Header Length : They are all UDP packets and have the same header length.
Differentiated Services Field : They are all UDP packets and are in the same class.
Protocol : They are all UDP packets.
Source Address : They are all sent from the same source.
Destination Address : They are all sent to the same destination.

Problem 8: Describe the pattern you see in the values in the Identification field of the IP datagrams being sent by your computer.
Answer The values in the identification field increment with each subsequent packet.

Topic 3-3 : Fragmentation

Problems

Problem 9: Find the first IP datagram containing the first part of the segment sent to 128.119.245.12 sent by your computer via the traceroute command to gaia.cs.umass.edu, after you specified that the traceroute packet length should be 3000. Has that segment been fragmented across more than one IP datagram?
Answer Yes, the segment is fragmented into 3 IP datagrams.

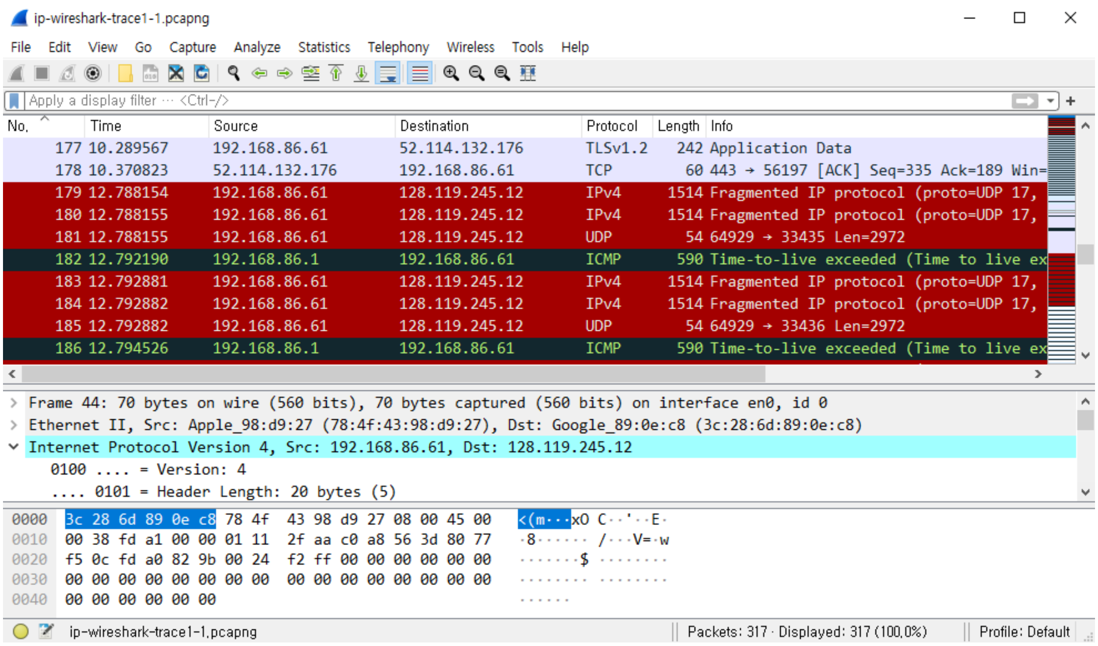


Figure 22: Problem 3-9’s screenshot :

Problem 10: What information in the IP header indicates that this datagram been fragmented?
Answer Since the flags more fragments bit is ‘Set’, we know that this datagram has been fragmented.

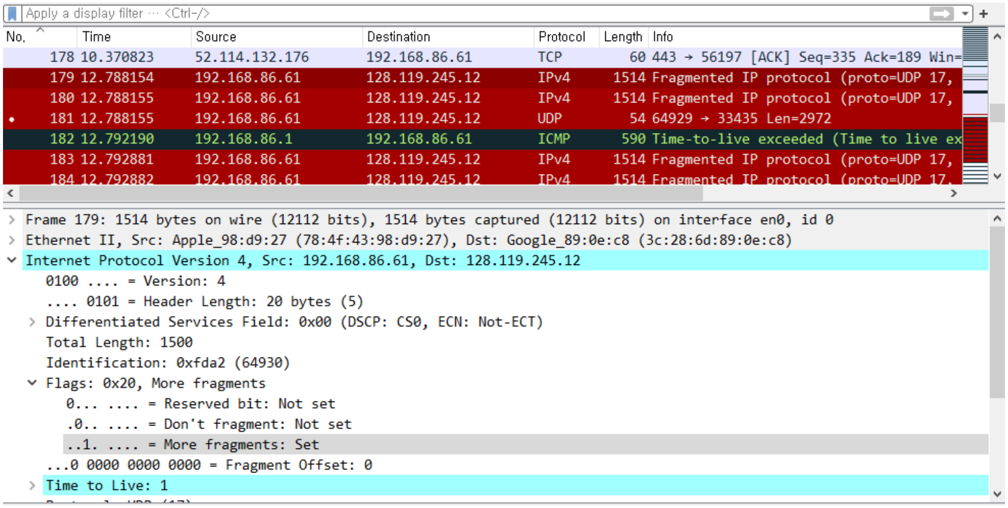


Figure 23: Problem 3-10's screenshot :

Problem 11: What information in the IP header for this packet indicates whether this is the first fragment versus a latter fragment?

Answer Since the fragment offset is 0, we know that this is the first fragment.

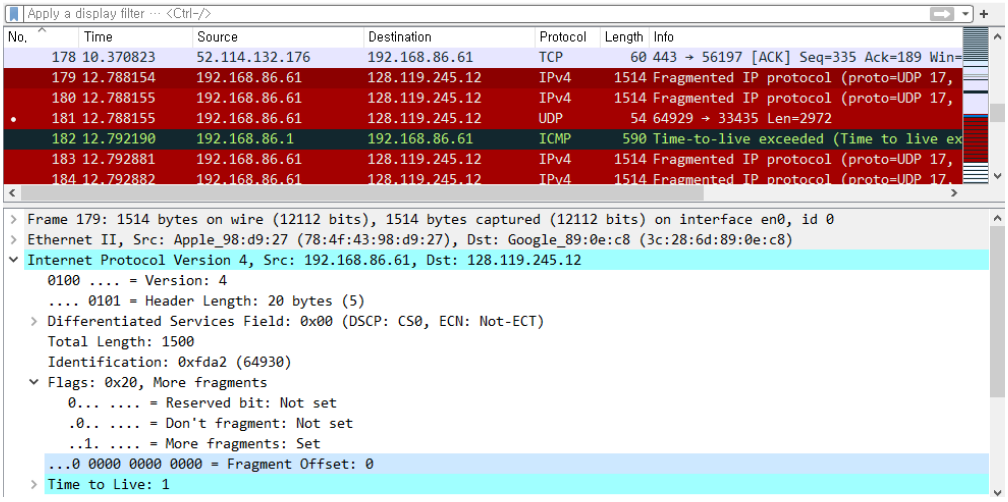


Figure 24: Problem 3-11's screenshot :

Problem 12: How many bytes are there in is this IP datagram (header plus payload)?

Answer 1500 bytes

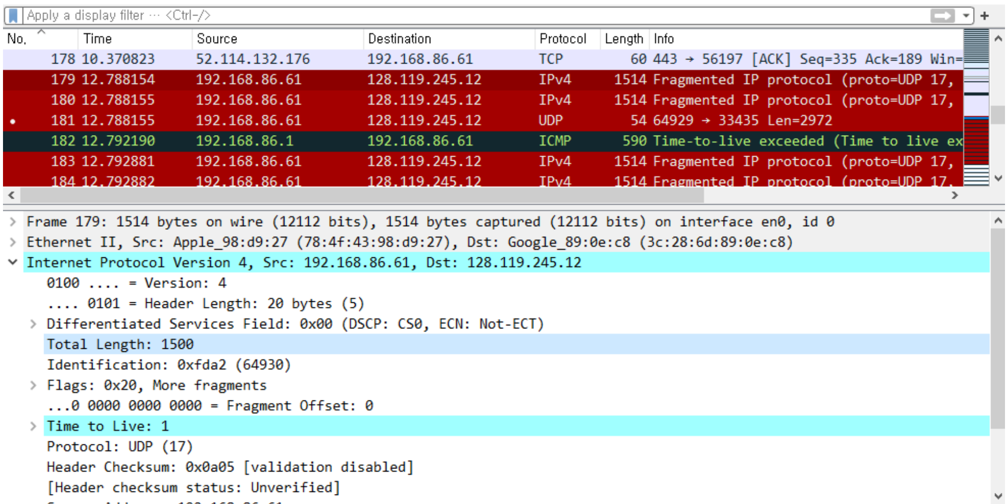


Figure 25: Problem 3-12's screenshot :

Problem 13: Now inspect the datagram containing the second fragment of the fragmented UDP segment. What information in the IP header indicates that this is not the first datagram fragment?

Answer Since the fragment offset is 1480, we know that this is not the first fragment.

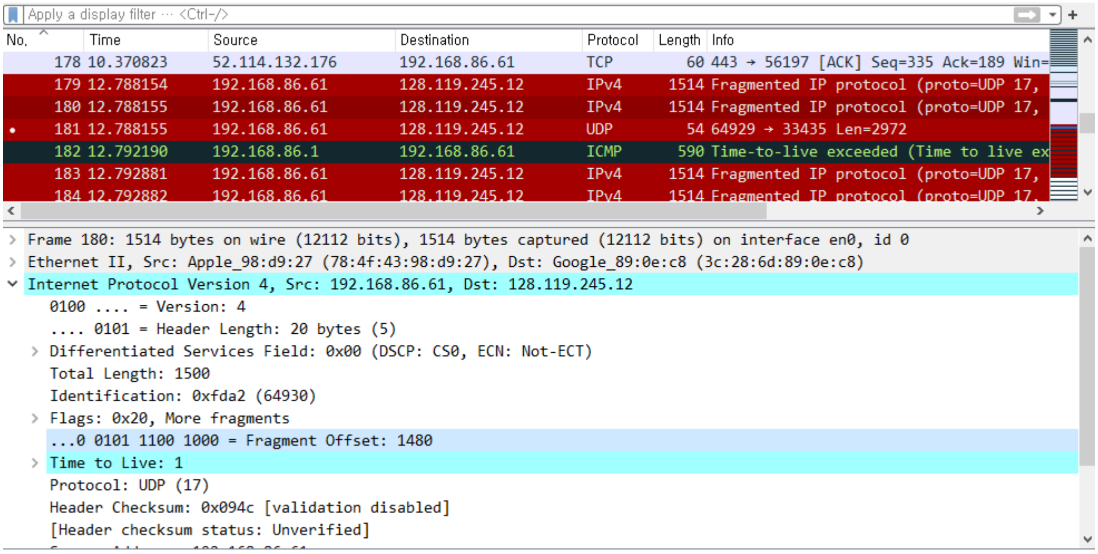
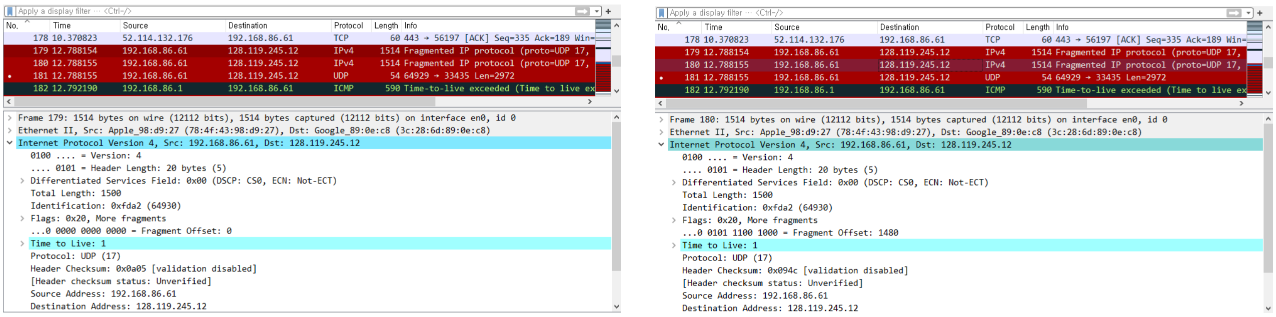


Figure 26: Problem 3-13's screenshot :

Problem 14: What fields change in the IP header between the first and second fragment?

Answer Fragment offset and Header checksum



(a) (b)

Figure 27: Problem 3-14's screenshot :