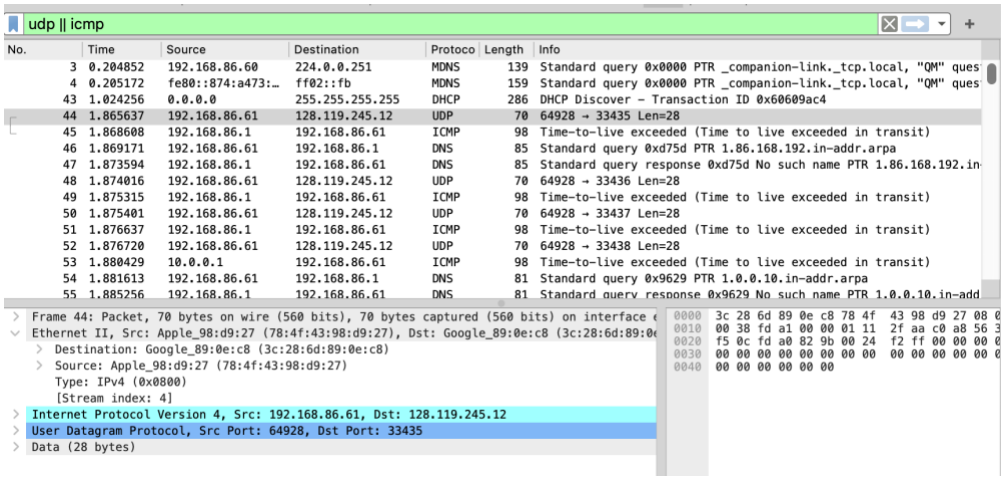
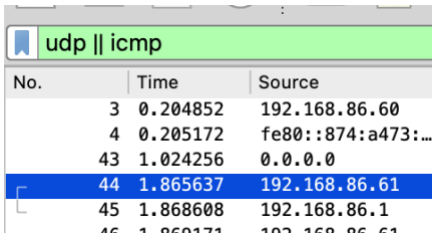
Catalina Cisneros

Lab: Wireshark IP v9.0



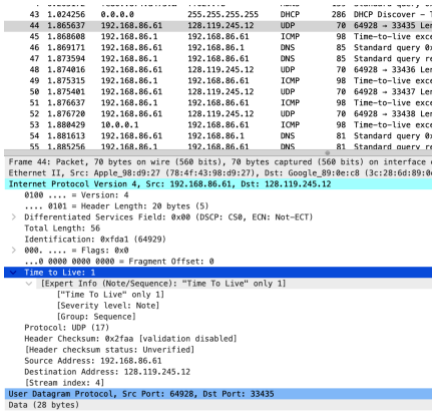
1. Select the first UDP segment sent by your computer via the traceroute command to gaia.cs.umass.edu. What is the IP address of your computer?

→ The IP address of my computer is 192.168.86.61



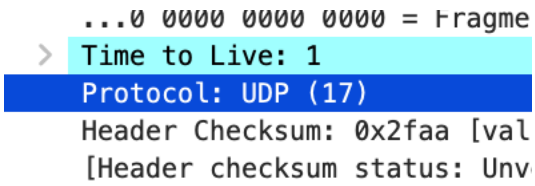
2. What is the value in the time to live (TTL) field in this IPv4 datagram's header?

→ The TTL value in the IPv4 header is 1.



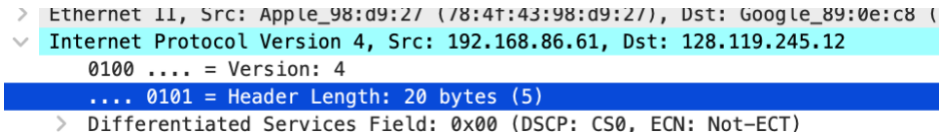
3. What is the value in the upper layer protocol field in this IPv4 datagram's header?

→ UDP (protocol 17)



4. How many bytes are in the IP header?

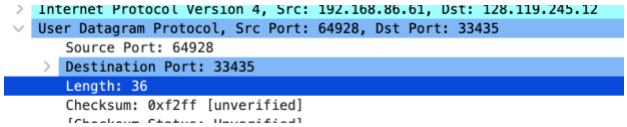
→ 20 bytes



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

→ 36 bytes

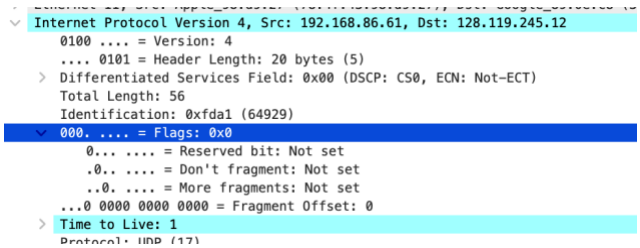
→ Total Length: 56 (IP header + payload) Header Length: 20 bytes → Payload = 56 − 20 = 36 bytes, which matches the UDP Length: 36 field.



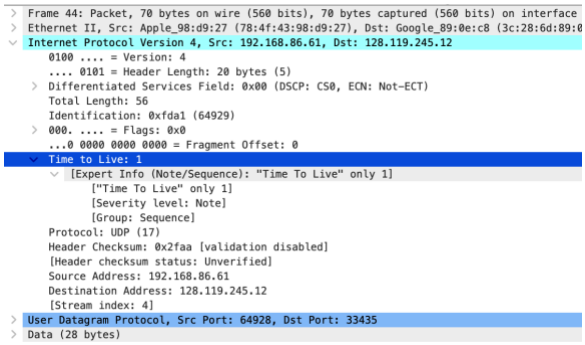
6. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented.

→ No, not fragmented.

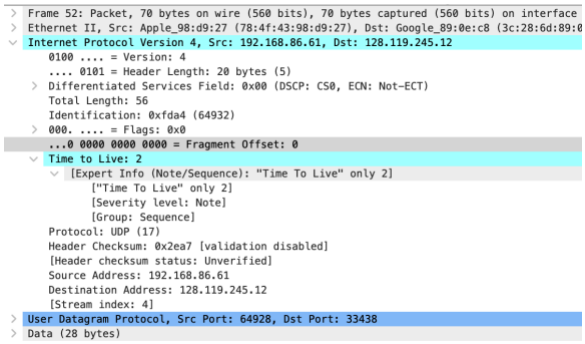
→ Flags: 0x0 means the More Fragments bit is not set. Fragment Offset: 0 → Both together indicate this datagram is a single, unfragmented IP packet.



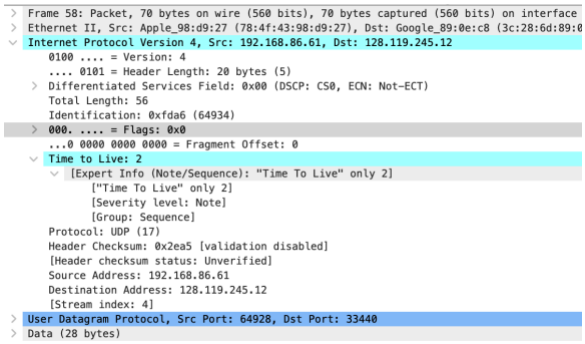
→ packet 44



→ packet 52



→ packet 58



7. Which fields in the IP datagram always change from one datagram to the next within this series of UDP segments? Why?

→ The fields that change from one datagram to the next are the TTL, the Identification field, the header checksum, and the UDP destination port. These change because traceroute sends each probe with a new TTL to discover the next hop, assigns a new Identification value for every datagram, recalculates the checksum based on the modified header, and varies the UDP destination port so it can match the returning ICMP messages to specific traceroute probes.

8. Which fields in this sequence of IP datagrams stay constant? Why?

→ The fields that stay constant across the sequence are the source IP address, destination IP address, IPv4 version, header length, differentiated services field, flags, fragment offset, the protocol (UDP), and the payload size. These remain unchanged because all packets are being sent from the same machine to the same destination using the same protocol settings, and none of these packets require fragmentation

9. Describe the pattern you see in the values in the Identification field of the IP datagrams being sent by your computer.

→ Packet 44: 0xfda1

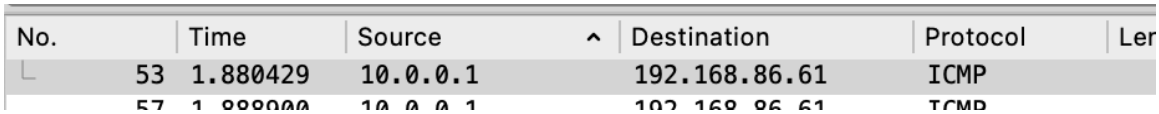
→ Packet 52: 0xfda4

→ Packet 58: 0xfda9

→ These values consistently increase as new packets are sent.

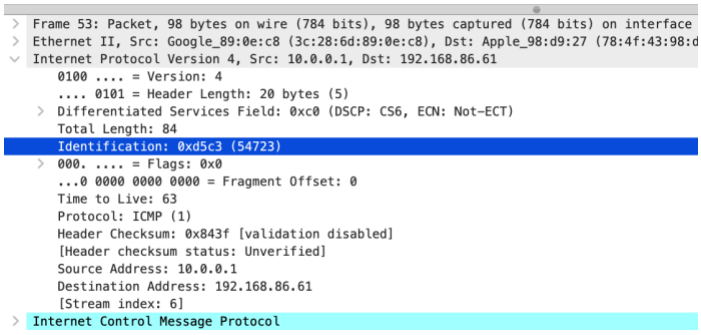
10. What is the upper layer protocol specified in the IP datagrams returned from the routers?

→ ICMP (1)

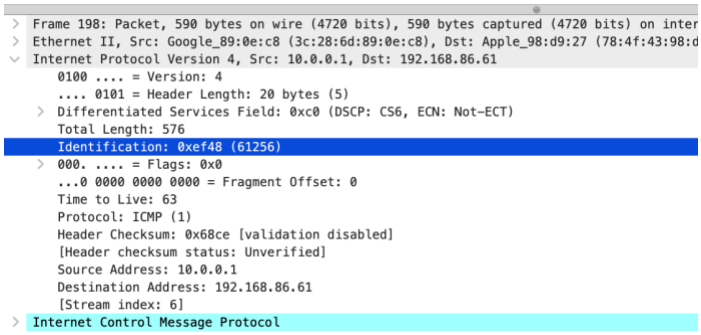


11. Are the values in the Identification fields similar in behavior to your answer to question 9?

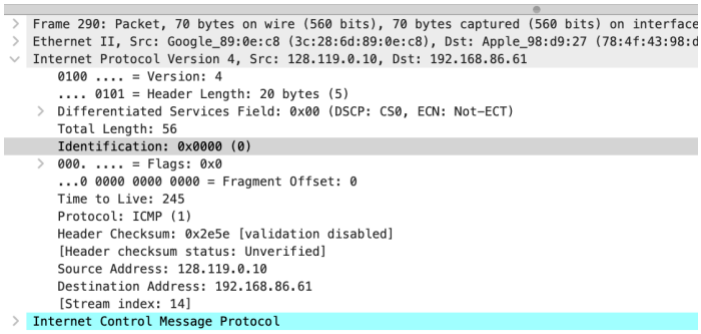
→ packet 53



→ packet 198



→ packet 290



→ No, the Identification values in the ICMP packets are not similar to the pattern seen in Question 9. In my outbound UDP packets (Q9), the Identification numbers increased sequentially. However, in the ICMP packets returned by different routers, the Identification numbers vary widely and do not follow a sequence

12. Are the values of the TTL fields similar across all of the ICMP packets from all routers?

→ No. The TTL values are not similar across the ICMP packets. Each ICMP packet comes from a different router, and every router sets its own initial TTL value when generating the ICMP "Time Exceeded" message. This means the TTL values differ from packet to packet

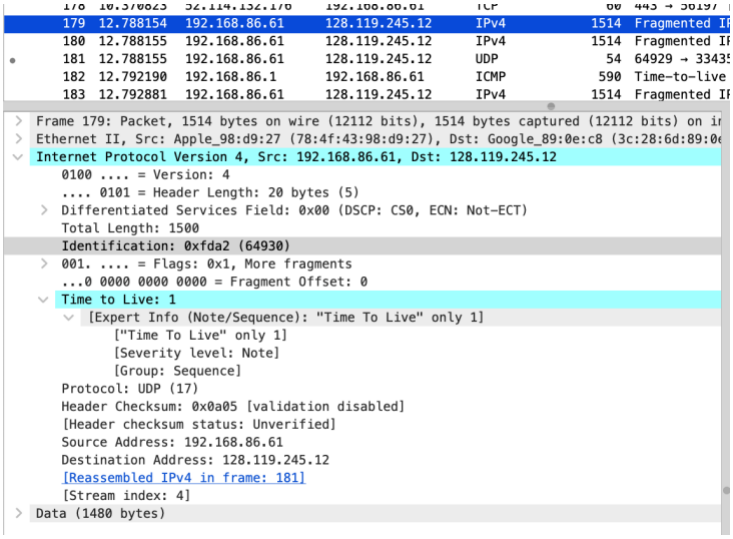
→ Packet 53: TTL = 63

→ Packet 198: TTL = 63

→ Packet 290: TTL = 245

13. Find the first IP datagram containing the first part of the 3000 byte segment. Has that segment been fragmented across more than one IP datagram?

→ Yes. It's split into multiple pieces. Packet 179 shows the same ID as the others and the "more fragments" flag is on.

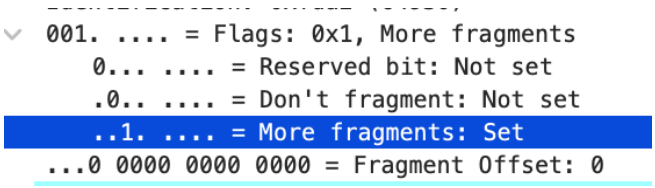


14. What information in the IP header indicates that this datagram has been fragmented?

→ The IP header has More Fragments = 1 and you can see the fragment offset field being used

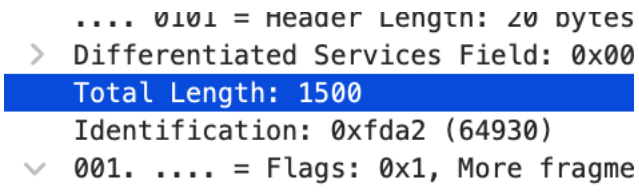
15. What information in the IP header indicates whether this is the first fragment versus a later fragment?

→ The fragment offset is 0, and more fragments is 1, so this is the first piece

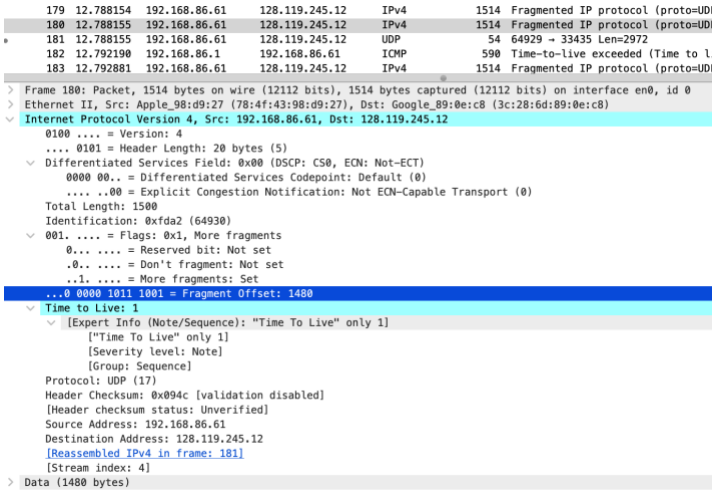


16. How many bytes are in this IP datagram (header plus payload)?

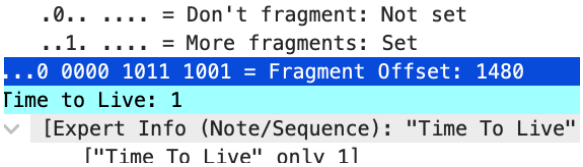
→ The Total Length is 1500 bytes, so this fragment is 1500 bytes total



17. Inspect the second fragment. What information indicates that this is not the first fragment?



→ Because the fragment offset is 1480, not 0. First fragments always start at offset 0

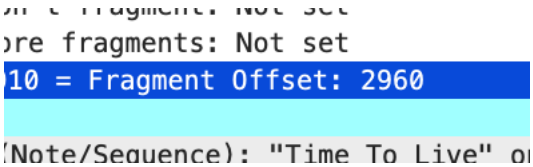


18. What fields change in the IP header between the first and second fragment?

→ The fragment offset changed (from 0 to 1480), and the header checksum changed. The total length stays the same because both fragments are the same size

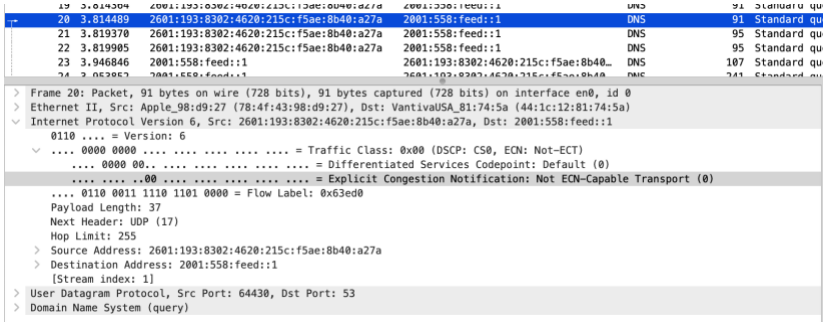
19. Find the third fragment. What information indicates this is the last fragment?

→ Because More Fragments is 0, and the fragment offset is 2960, which means it's the final chunk of the original packet



20. What is the IPv6 source address of the DNS AAAA request?

→ The source IPv6 address is 2601:193:8302:4620:215c:f5ae:8b40:a27a

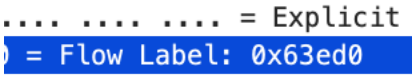


21. What is the IPv6 destination address of this datagram?

→The destination IPv6 address is 2001:558:feed::1

22. What is the value of the flow label for this datagram?

→ The flow label is 0x63ed0

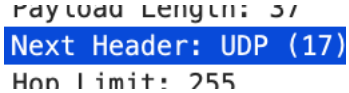


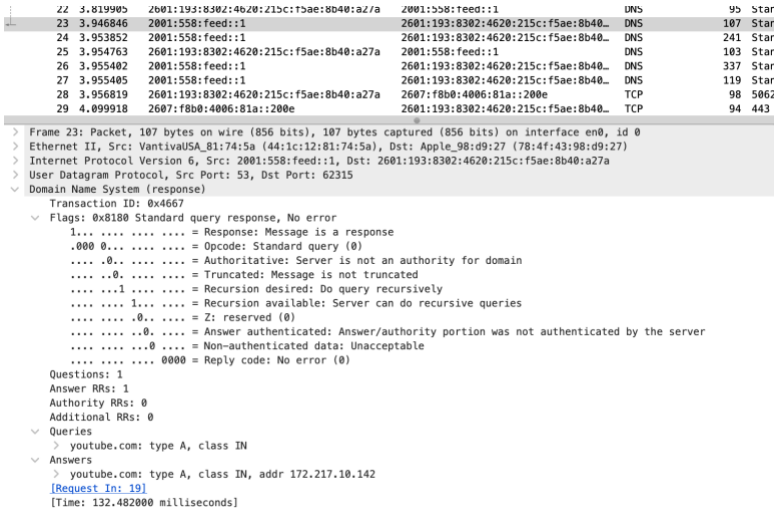
23. How much payload data is carried in this datagram?

→ The payload length is 37 bytes

24. What is the upper layer protocol for this datagram's payload?

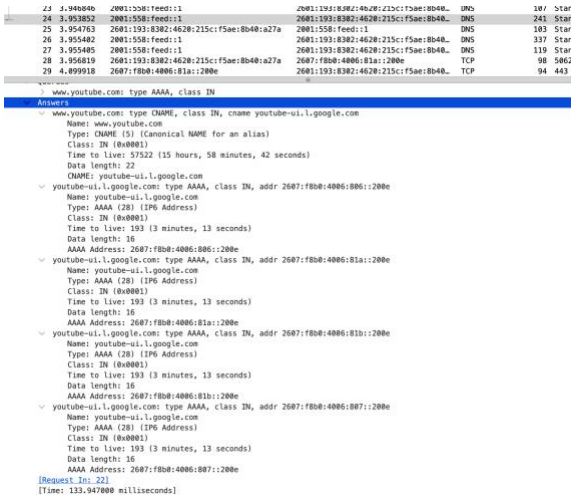
→ The upper layer protocol is UDP (17)





25. How many IPv6 addresses are returned in the response to this AAAA request?

→ It returned 4 IPv6 addresses.



26. What is the first IPv6 address returned for youtube.com?

→ The first AAAA address is: 2607:f8b0:4006:806::200e