Step 1. Capture a trace

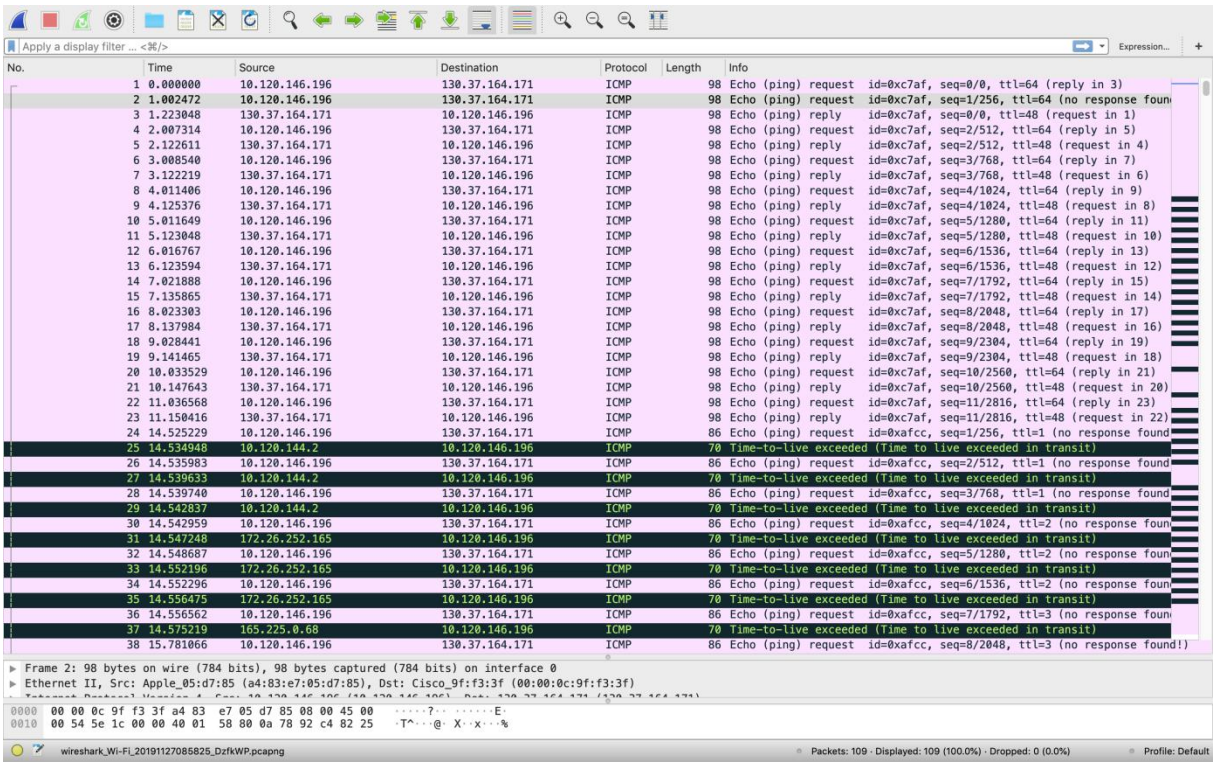
1) Find a URL, I am using this one, www.cs.vu.nl
2) I can successfully ping it.

```
➜  ~ ping www.cs.vu.nl
PING papac022.vu.nl (130.37.164.171): 56 data bytes
64 bytes from 130.37.164.171: icmp_seq=0 ttl=48 time=110.971 ms
64 bytes from 130.37.164.171: icmp_seq=1 ttl=48 time=113.431 ms
64 bytes from 130.37.164.171: icmp_seq=2 ttl=48 time=107.284 ms
64 bytes from 130.37.164.171: icmp_seq=3 ttl=48 time=110.297 ms
^C
--- papac022.vu.nl ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 107.284/110.496/113.431/2.191 ms
```

3) Trace the route using traceroute command

```
➜  ~ traceroute -I www.cs.vu.nl
traceroute to papac022.vu.nl (130.37.164.171), 64 hops max, 72 byte packets
 1  10.120.144.2 (10.120.144.2)  4.330 ms  5.315 ms  4.750 ms
 2  172.26.252.165 (172.26.252.165)  4.460 ms  4.430 ms  4.520 ms
 3  165.225.0.68 (165.225.0.68)  19.220 ms  18.016 ms  19.841 ms
 4  165.225.0.2 (165.225.0.2)  19.671 ms  24.560 ms  22.325 ms
 5  64.125.46.72.network.zip.zayo.com (64.125.46.72)  19.659 ms  20.017 ms  20.445 ms
 6  * * ae10.cs3.ord2.us.zip.zayo.com (64.125.28.176)  108.043 ms
 7  ae3.cs1.lga5.us.eth.zayo.com (64.125.29.208)  102.261 ms  102.088 ms  100.409 ms
 8  ae5.cs1.lhr11.uk.eth.zayo.com (64.125.29.127)  102.574 ms  101.990 ms  101.514 ms
 9  ae27.mpr2.lhr2.uk.zip.zayo.com (64.125.30.237)  102.959 ms  101.849 ms  105.847 ms
10  ae11.mpr1.lhr15.uk.zip.zayo.com (64.125.30.53)  104.844 ms  101.840 ms  163.946 ms
11  * * *
12  vu-router.customer.surf.net (145.145.20.58)  111.102 ms  113.128 ms  109.145 ms
13  130.37.6.94 (130.37.6.94)  109.174 ms  107.589 ms  109.298 ms
14  130.37.6.98 (130.37.6.98)  201.790 ms  112.690 ms  107.916 ms
15  130.37.250.126 (130.37.250.126)  108.071 ms  109.389 ms  108.938 ms
16  130.37.164.171 (130.37.164.171)  109.916 ms  108.976 ms  108.357 ms
```

4) Set up the filter to be ‘icmp’, then repeat the ping command in 1) step and traceroute command in 2) step:



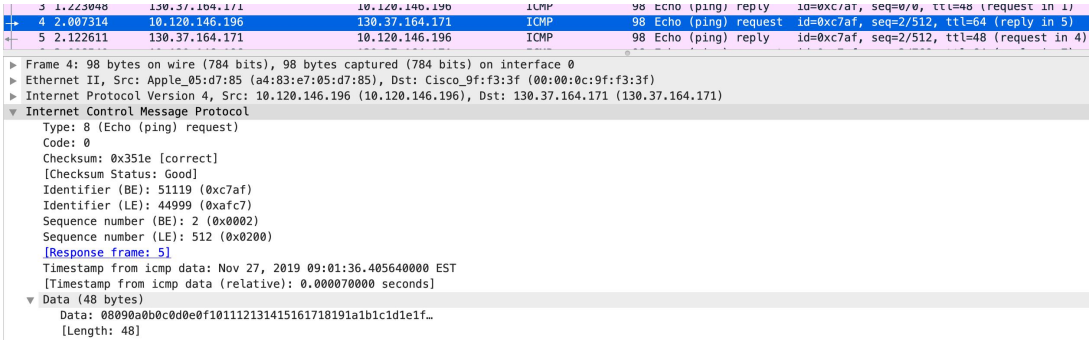
Step 2. Echo (ping) Packets

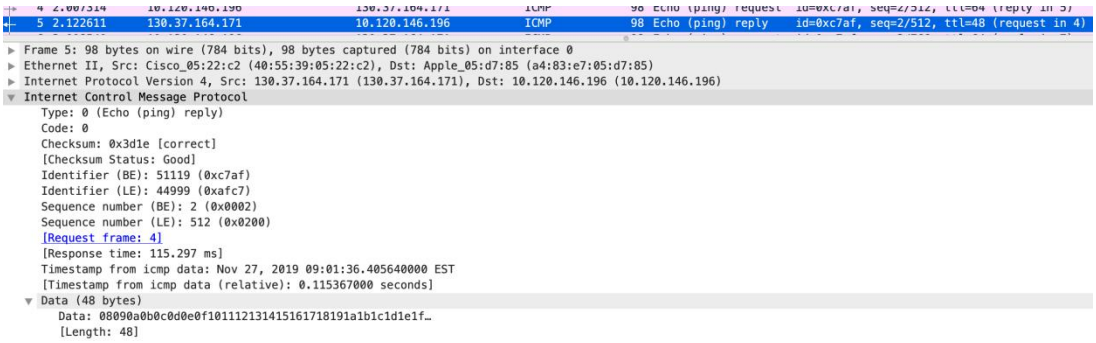
Answer the following questions to demonstrate your understanding of ICMP echo messages:

1. What are the Type/Code values for an ICMP echo request and echo reply packet, respectively?

--> Echo Request: Type is 8, Code is 0

Echo Reply: Type is 0, Code is 0





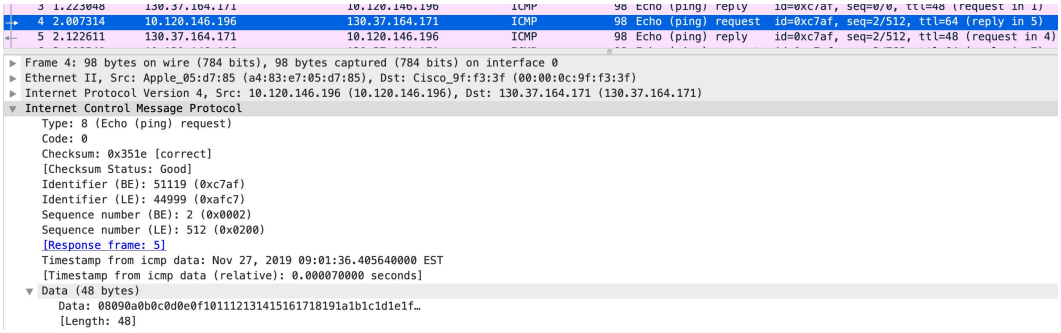
2. How do the Identifier and Sequence Number compare for an echo request and the corresponding echo reply?

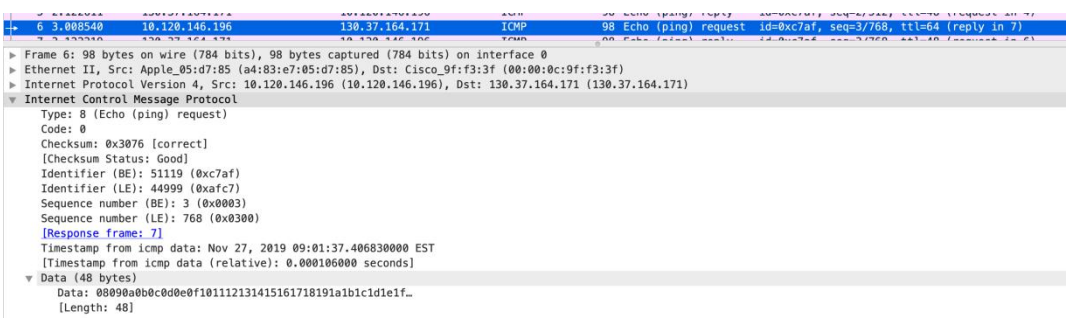
--> Both Identifier and Sequence Number are the same in echo request and its corresponding echo reply.

3. How do the Identifier and Sequence Number compare for successive echo request packets?

Identified stays the same.

--> Sequence Number increase by 1.





4. Is the data in the echo reply the same as in the echo request or different?

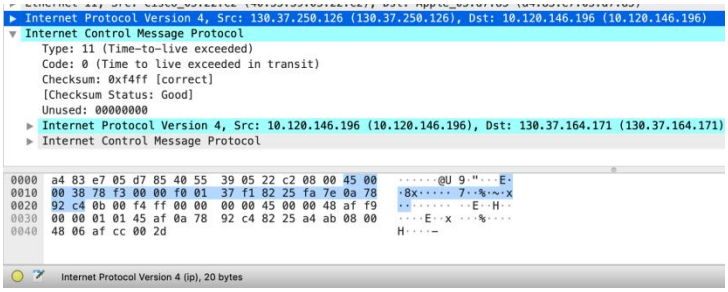
--> The same.

Step 3. TTL Exceeded (traceroute) Packets

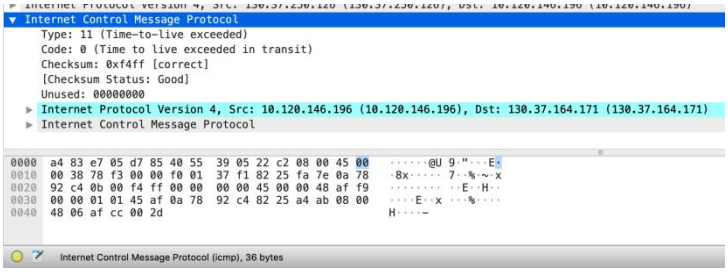
Draw a picture of one ICMP TTL Exceeded packet to make sure that you understand its nested structure.

IP Header fields	+ Type (11)	+ Code(0)	+ Checksum	+ Unused	+ ICMP Payload
28 bytes	1 byte	1 byte	2 bytes	4 bytes	28 bytes
					[IP Header + ICMP Header]
IP Header	ICMP Header				20 bytes 8 bytes

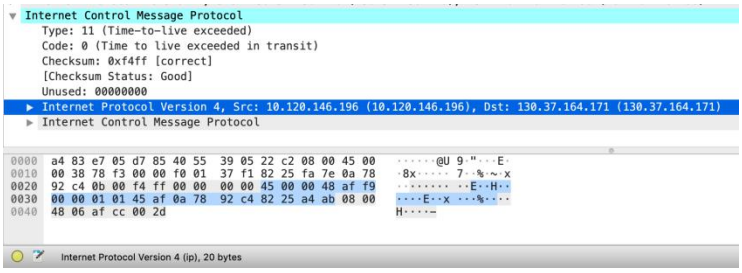
IP Header --> 20 bytes



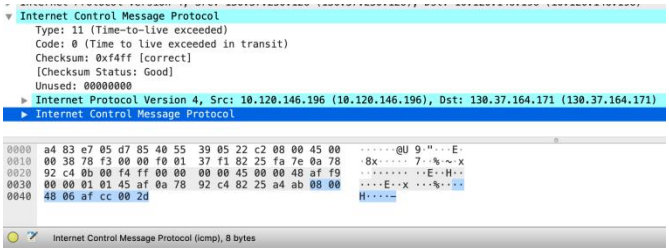
ICMP Header --> 36 bytes



ICMP Payload1 (IP Header) --> 20 bytes



ICMP Payload2 (ICMP Header) --> 8 bytes



Answer the following questions:

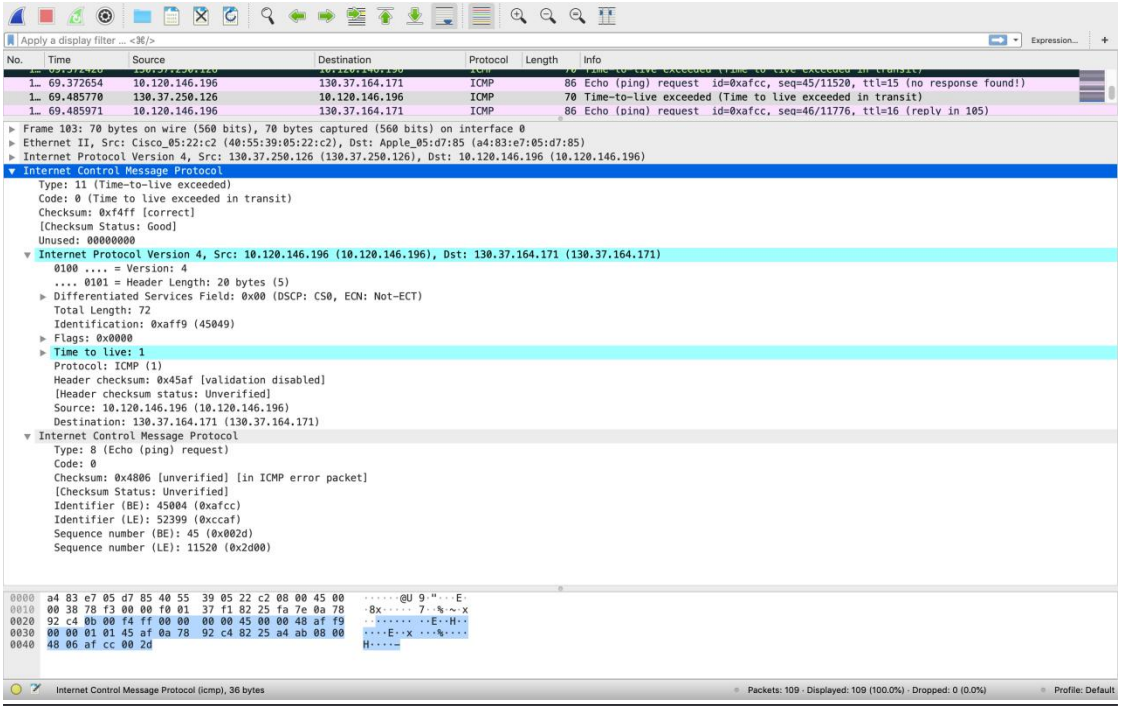
1. What is the Type/Code value for an ICMP TTL Exceeded packet?

--> Type is 11, Code is 0

2. Say how the receiver can safely find and process all the ICMP fields if it does not know ahead of time what kind of ICMP message to expect. The potential issue, as you have probably noticed, is that different ICMP messages can have different formats. For instance, Echo has Sequence and Identifier fields while TTL Exceeded does not.
--> Receiver can parse the Type field in the ICMP header, and since all the ICMP messages all starts will Type, Code and Checksum. In this way, receiver will know what field coming next.

3. How long is the ICMP header of a TTL Exceeded packet? Select different parts of the header in Wireshark to see how they correspond to the bytes in the packet.
--> It is 8 bytes, 1 byte Type + 1 byte Code + 2 bytes Checksum + 4 bytes Unused

4. The ICMP payload contains an IP header. What is the TTL value in this header? Explain why it has this value. Guess what it will be before you look!
--> The TTL value in the header is 1. The value is 1 because the TTL Exceeded packet is triggered when the TTL value decreases by 1 and reach 0, therefore must be discarded.



Step 4. Internet Paths

By looking at the details of the packets, answer the following questions:
1. How does your computer (the source) learn the IP address of a router along the path from a TTL exceeded packet? Say where on this packet the IP address is found. You might proceed by looking at an echo packet to see the source and destination IP addresses. The routers along the path will have a different IP address, and this address will be present on the TTL Exceeded packet. If you are unsure, you can examine the traceroute text output to see the IP addresses of routers and look for these addresses on the TTL Exceeded packets.
--> The IP source field in the IP header of the TTL exceeded packet is the IP address of the router, this is because the packet is generated by the router.

```
27 14.539633   10.120.144.2      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
28 14.539740   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=3/768, ttl=1 (no response found!)
29 14.542837   10.120.144.2      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)

▶ Frame 27: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
▶ Ethernet II, Src: Cisco_eb:f0:42 (04:c5:a4:eb:f0:42), Dst: Apple_05:d7:85 (a4:83:e7:05:d7:85)
▶ Internet Protocol Version 4, Src: 10.120.144.2 (10.120.144.2), Dst: 10.120.146.196 (10.120.146.196)
▼ Internet Control Message Protocol
    Type: 11 (Time-to-live exceeded)
    Code: 0 (Time to live exceeded in transit)
    Checksum: 0xf4ff [correct]
    [Checksum Status: Good]
    Unused: 00000000
  ▶ Internet Protocol Version 4, Src: 10.120.146.196 (10.120.146.196), Dst: 130.37.164.171 (130.37.164.171)
  ▼ Internet Control Message Protocol
      Type: 8 (Echo (ping) request)
      Code: 0
      Checksum: 0x4831 [unverified] [in ICMP error packet]
      [Checksum Status: Unverified]
      Identifier (BE): 45004 (0xafcc)
      Identifier (LE): 52399 (0xccaf)
      Sequence number (BE): 2 (0x0002)
      Sequence number (LE): 512 (0x0200)
```

2. How many times is each router along the path probed by traceroute? Look at the TTL Exceeded responses and see if you can discern a pattern.

--> As we could, traceroute probe each router for 3 times.

```
25 14.534948   10.120.144.2      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
26 14.535983   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=2/512, ttl=1 (no response found!)
27 14.539633   10.120.144.2      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
28 14.539740   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=3/768, ttl=1 (no response found!)
29 14.542837   10.120.144.2      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
30 14.542959   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=4/1024, ttl=2 (no response found!)
31 14.547248   172.26.252.165    10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
32 14.548687   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=5/1280, ttl=2 (no response found!)
33 14.552196   172.26.252.165    10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
34 14.552296   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=6/1536, ttl=2 (no response found!)
35 14.556475   172.26.252.165    10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
36 14.556562   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=7/1792, ttl=3 (no response found!)
37 14.575219   165.225.0.68      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
38 15.781066   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=8/2048, ttl=3 (no response found!)
39 15.798208   165.225.0.68      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
40 15.798437   10.120.146.196    130.37.164.171    ICMP   86 Echo (ping) request  id=0xafcc, seq=9/2304, ttl=3 (no response found!)
41 15.817575   165.225.0.68      10.120.146.196    ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
```

3. How does your computer (the source) craft an echo request packet to find (by eliciting a TTL Exceeded response) the router N hops along the path towards the destination? Describe the key attributes of the echo request packet. The echo request packets sent by traceroute are probing successively more distant routers along the path. You can look at these packets and see how they differ when they elicit responses from different routers.

--> The IP header holds the source IP and the destination IP, together with a TTL which has a value N. The key attribute of the echo request packet is the TTL field. The routers will decrease the TTL along the path which will lead to TTL reaching 0 N hops away from the source. Then that router will send a TTL exceeded message to the source. In this way the source could find the router N hops along the path towards the destination.

Traceroute

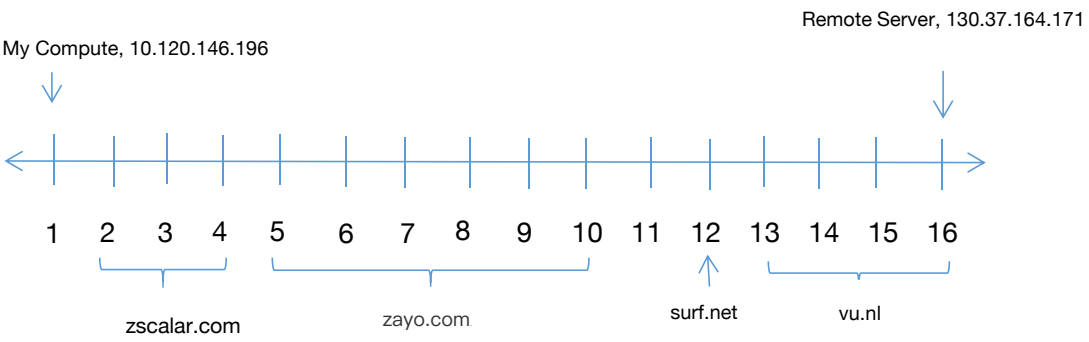


Fig. Internet Paths