ENGR 3321 - Lab 6: Wireshark IP

28 October 2022

Nathan Cauwet

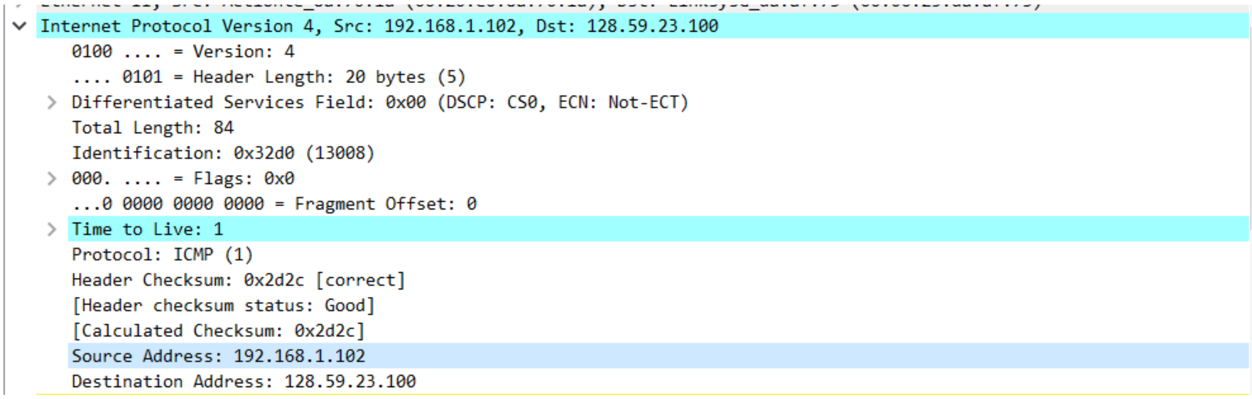
873271826

Note: I had to use the gaia.cs.umass.edu traces because my trace was polluted with packets from programs that I was not able to close at the time.

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?

192.168.1.102



```
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.59.23.100
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 84
    Identification: 0x32d0 (13008)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
  > Time to Live: 1
    Protocol: ICMP (1)
    Header Checksum: 0x2d2c [correct]
    [Header checksum status: Good]
    [Calculated Checksum: 0x2d2c]
    Source Address: 192.168.1.102
    Destination Address: 128.59.23.100
```

2. *Within the IP packet header, what is the value in the upper layer protocol field?*

ICMP (code 1) (0x01 in the hex dump)

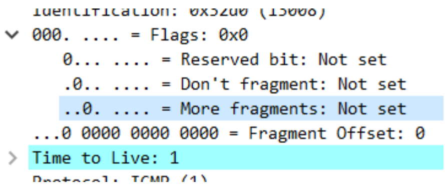
3. *How many bytes are in the IP header? How many bytes are in the payload of the IP datagram? Explain how you determined the number of payload bytes.*

Bytes in packet = 84, header bytes = 20, therefore payload bytes = 84-20 = 56

I subtracted the number of header bytes from the total length of the packet.

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

The ‘more fragments’ bit (flag) is not set, therefore the datagram was **not fragmented**.



```
Identification: 0x32d0 (13008)
000. .... = Flags: 0x0
    0... .... = Reserved bit: Not set
    .0.. .... = Don't fragment: Not set
    ..0. .... = More fragments: Not set
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 1
Protocol: ICMP (1)
```

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?*

The Identification field and the Time to Live change from one datagram to the next. From what I see, they both increment by 1 with each iteration. The checksum of course changes from one to the next, it appears to decrease by 0x1 with each iteration.

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

The first half of the Identity field stays constant (more detail in Q7).

The header, the Differentiated Services Field, the flags, version, source ip and destination ip all remain constant throughout each repetition.

The source and destination IPs must remain the same, if they do not then it’s not running a trace at all. The checksum must remain the same so that a new authorization value does not have to generate for every ping. The header must remain the same so that the server can recognize the request.

For the “Time to Live” field, I believe only the count/iterator is changing. There is an additional ‘note’ from wireshark on any value for this field below 5.

7. *Describe the pattern you see in the values in the Identification field of the IP datagram.*

The Indentification field increases by 0x1 (1) for every ping request. The first ping request has the Ident. Field of 0x32d0. The 2 far left hex bits stay the same, so in this case for each iteration of the ping, the value is 0x32__ where __ is 0xd0 + count.

Next (with the packets still sorted by source address) find the series of ICMP TTL- exceeded replies sent to your computer by the nearest (first hop) router.

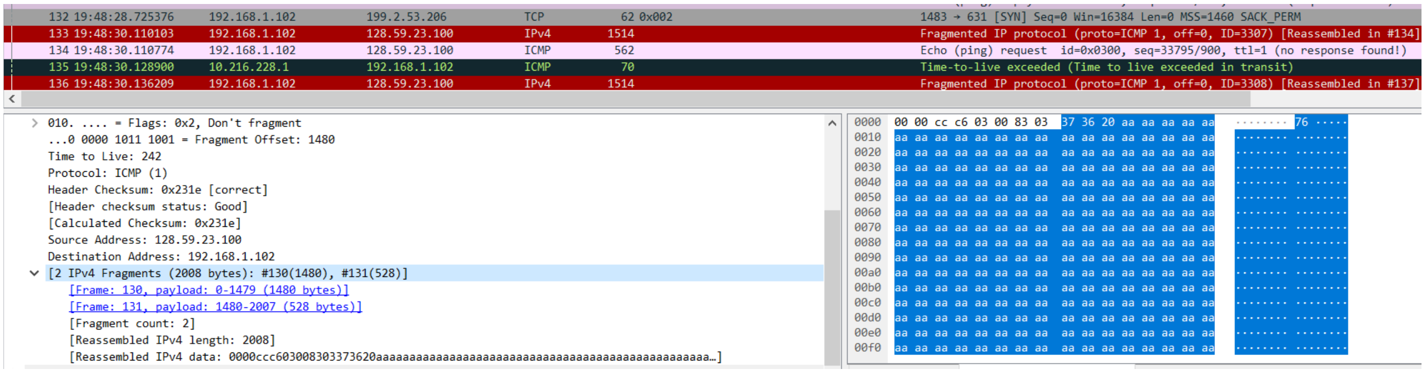
8. What is the value in the Identification field and the TTL field?

Identification is 0x0000 (0) and the TTL is 246

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 stays the same but the TTL does not. This is in order to group the packets into one series (identification being the same to group the responses). The TTL does not stay the same because some fields still differ between the different responses, even though identification does not.

Sort the packet listing according to time again by clicking on the Time column.



```
132 19:48:28.725376   192.168.1.102   199.2.53.206    TCP     62 0x002   1483 → 631 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
133 19:48:30.110103   192.168.1.102   128.59.23.100   IPv4  1514         Fragmented IP protocol (proto=ICMP 1, off=0, ID=3307) [Reassembled in #134]
134 19:48:30.110774   192.168.1.102   128.59.23.100   ICMP   562         Echo (ping) request  id=0x0300, seq=33795/900, ttl=1 (no response found!)
135 19:48:30.128900   10.216.228.1    192.168.1.102   ICMP    70         Time-to-live exceeded (Time to live exceeded in transit)
136 19:48:30.136209   192.168.1.102   128.59.23.100   IPv4  1514         Fragmented IP protocol (proto=ICMP 1, off=0, ID=3308) [Reassembled in #137]
```

```
> 010. .... = Flags: 0x2, Don't fragment
  ...0 0000 1011 1001 = Fragment Offset: 1480
  Time to Live: 242
  Protocol: ICMP (1)
  Header Checksum: 0x231e [correct]
  [Header checksum status: Good]
  [Calculated Checksum: 0x231e]
  Source Address: 128.59.23.100
  Destination Address: 192.168.1.102
v [2 IPv4 Fragments (2008 bytes): #130(1480), #131(528)]
     [Frame: 130, payload: 0-1479 (1480 bytes)]
     [Frame: 131, payload: 1480-2007 (528 bytes)]
     [Fragment count: 2]
     [Reassembled IPv4 length: 2008]
     [Reassembled IPv4 data: 0000ccc603008303373620aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa…]
```

```
0000  00 00 cc c6 03 00 83 03  37 36 20 aa aa aa aa aa   ········ 76 ·····
0010  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0020  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0030  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0040  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0050  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0060  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0070  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0080  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
0090  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
00a0  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
00b0  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
00c0  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
00d0  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
00e0  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
00f0  aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa   ········ ········
```

10. Find the first ICMP Echo Request message that was sent by your computer after you changed the Packet Size in pingplotter to be 2000. Has that message been fragmented across more than one IP datagram? [Note: if you find your packet has not been fragmented, you should download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the ip- ethereal-trace-1packet trace. If your computer has an Ethernet interface, a packet size of 2000 should cause fragmentation.3]

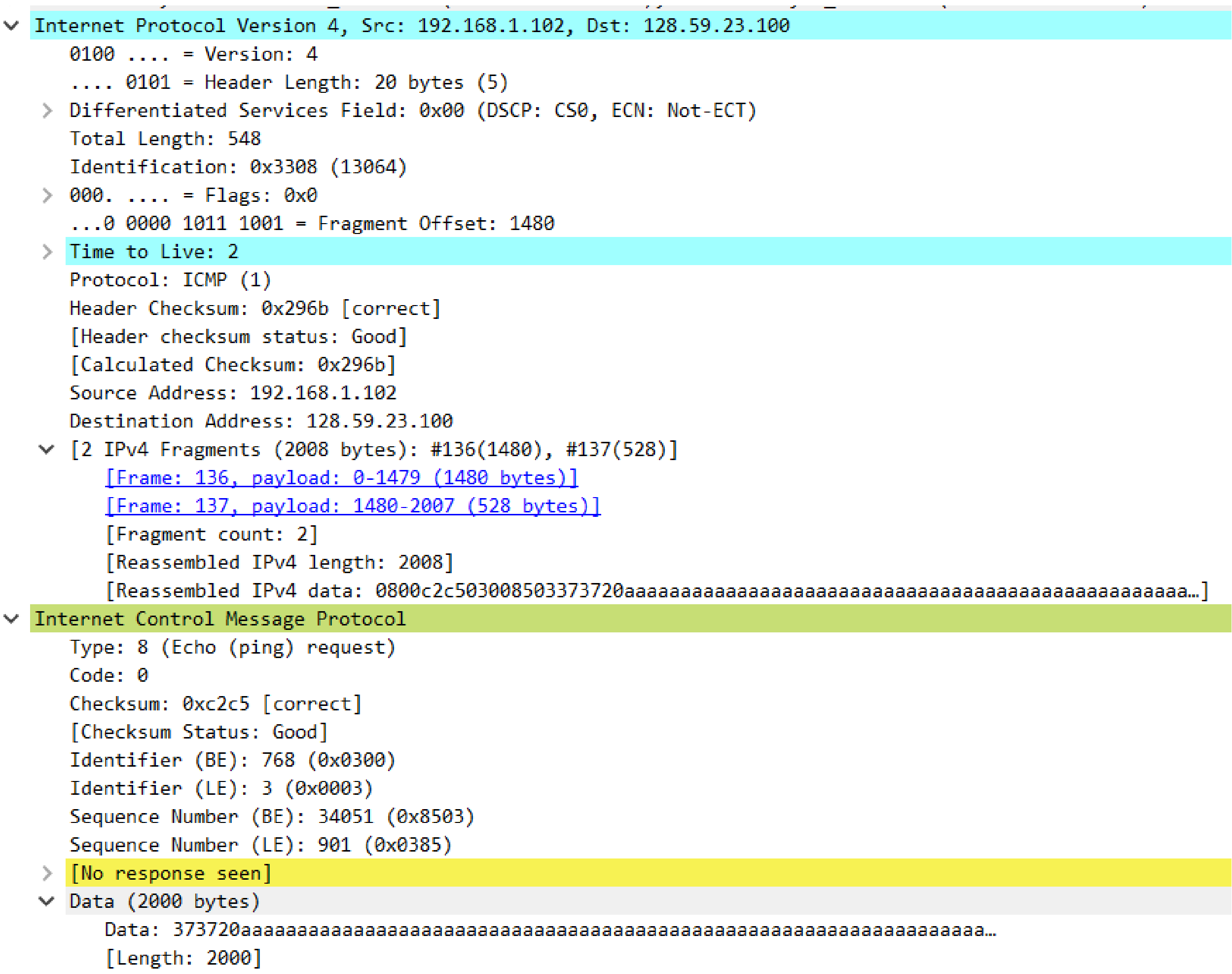
Yes it has been fragmented into 2 IP datagrams. The first is 1480 bytes and the 2nd is 528 bytes.

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 flag 0x1 is set which is the bit to set the 'More Fragments' flag. The TTL (equals 1 in this case) and the 'more fragments flag' indicate that this is the first fragment rather than a later one. The total length of the IP datagram is 1500 bytes.

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 IP header indicates a Fragment Offset of 1480 (the size of the first fragment) and the total length is 548



```
v Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.59.23.100
     0100 .... = Version: 4
     .... 0101 = Header Length: 20 bytes (5)
   > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
     Total Length: 548
     Identification: 0x3308 (13064)
   > 000. .... = Flags: 0x0
     ...0 0000 1011 1001 = Fragment Offset: 1480
   > Time to Live: 2
     Protocol: ICMP (1)
     Header Checksum: 0x296b [correct]
     [Header checksum status: Good]
     [Calculated Checksum: 0x296b]
     Source Address: 192.168.1.102
     Destination Address: 128.59.23.100
   v [2 IPv4 Fragments (2008 bytes): #136(1480), #137(528)]
        [Frame: 136, payload: 0-1479 (1480 bytes)]
        [Frame: 137, payload: 1480-2007 (528 bytes)]
        [Fragment count: 2]
        [Reassembled IPv4 length: 2008]
        [Reassembled IPv4 data: 0800c2c503008503373720aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa…]
v Internet Control Message Protocol
     Type: 8 (Echo (ping) request)
     Code: 0
     Checksum: 0xc2c5 [correct]
     [Checksum Status: Good]
     Identifier (BE): 768 (0x0300)
     Identifier (LE): 3 (0x0003)
     Sequence Number (BE): 34051 (0x8503)
     Sequence Number (LE): 901 (0x0385)
   > [No response seen]
   v Data (2000 bytes)
        Data: 373720aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa…
        [Length: 2000]
```

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

The header checksum, the TTL, and the length change. The flag field changes to 0 because this is the last fragment.

Now find the first ICMP Echo Request message that was sent by your computer after you changed the Packet Size in pingplotter to be 3500.

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

3 datagrams were created from the original

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

The fragment offset is different in each (so that the fragments can be patched back together without data loss). The checksum changes of course.

In the last fragment, the length and the flag value are different. The first 2 fragments maxed out the length (1500-headerBytes = 1480 bytes) but the 3rd just needed to finish the remainer (568 bytes). The flag changes to 0 on the last fragment to indicate that the fragmentation process is done.