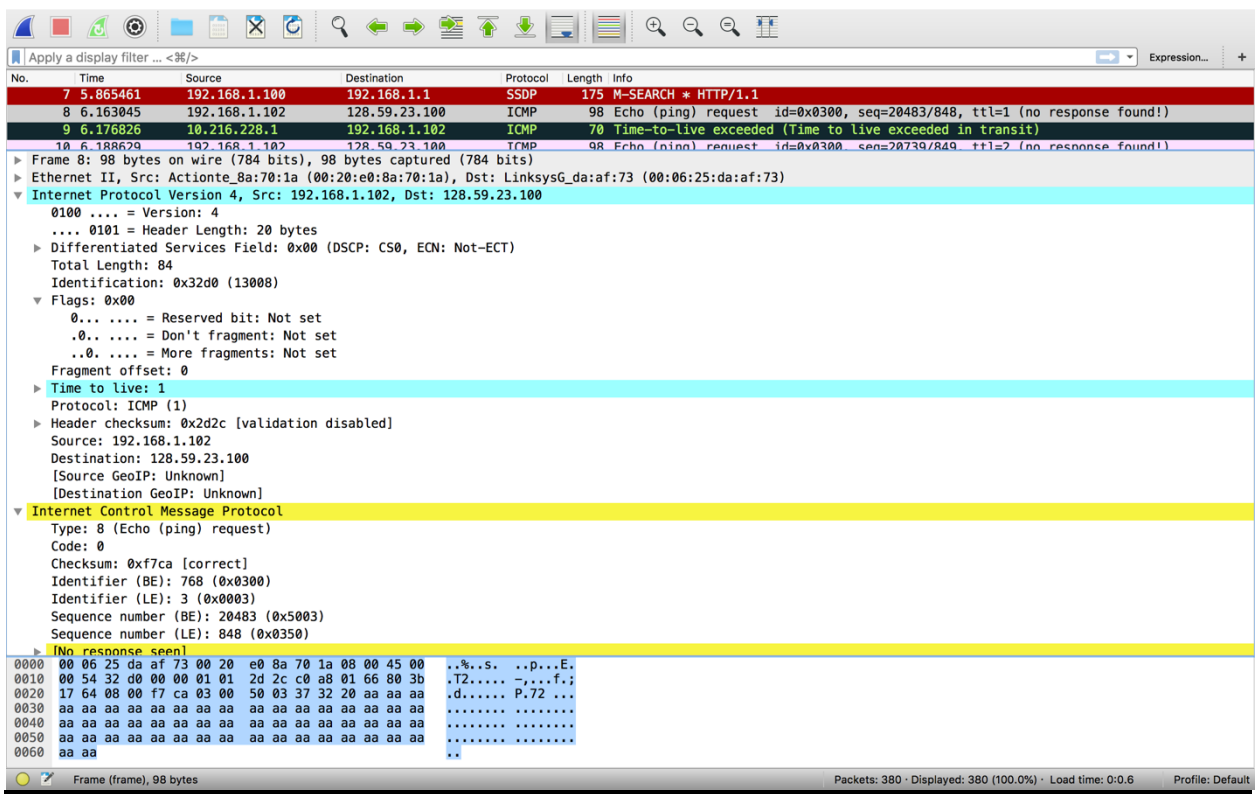
Farhad Ahmed
Computer Networking
11/08/18

WireShark Lab: IP

Running ifconfig:

```
farhads-MacBook-Pro:~ Farhad_Ahmed$ ipconfig
usage: ipconfig <command> <args>
where <command> is one of waitall, getifaddr, ifcount, getoption, getpacket, getv6packet, set, setverbose
farhads-MacBook-Pro:~ Farhad_Ahmed$ ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
        nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
XHC20: flags=0<> mtu 0
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether ac:bc:32:91:a2:af
        inet6 fe80::1cef:6e2b:c794:2a4%en0 prefixlen 64 secured scopeid 0x5
        inet6 2604:2000:6aa5:4500:42f:6470:410b:b373 prefixlen 64 autoconf secured
        inet6 2604:2000:6aa5:4500:ded:14d2:7e19:68c1 prefixlen 64 autoconf temporary
        inet 192.168.0.8 netmask 0xffffff00 broadcast 192.168.0.255
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
p2p0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 2304
        ether 0e:bc:32:91:a2:af
        media: autoselect
        status: inactive
awdl0: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1484
        ether fa:a3:79:0f:85:80
        inet6 fe80::f8a3:79ff:fe0f:8580%awdl0 prefixlen 64 scopeid 0x7
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
en1: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
        options=60<TSO4,TSO6>
        ether 4a:00:02:92:f1:80
        media: autoselect <full-duplex>
        status: inactive
en2: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
        options=60<TSO4,TSO6>
        ether 4a:00:02:92:f1:81
        media: autoselect <full-duplex>
        status: inactive
```

```
bridge0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        options=63<RXCSUM,TXCSUM,TSO4,TSO6>
        ether 4a:00:02:92:f1:80
        Configuration:
                id 0:0:0:0:0:0 priority 0 hellotime 0 fwddelay 0
                maxage 0 holdcnt 0 proto stp maxaddr 100 timeout 1200
                root id 0:0:0:0:0:0 priority 0 ifcost 0 port 0
                ipfilter disabled flags 0x2
        member: en1 flags=3<LEARNING,DISCOVER>
                ifmaxaddr 0 port 8 priority 0 path cost 0
        member: en2 flags=3<LEARNING,DISCOVER>
                ifmaxaddr 0 port 9 priority 0 path cost 0
        nd6 options=201<PERFORMNUD,DAD>
        media: <unknown type>
        status: inactive
utun0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 2000
        inet6 fe80::98a3:bf24:a130:cfaa%utun0 prefixlen 64 scopeid 0xb
        nd6 options=201<PERFORMNUD,DAD>
utun1: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1380
        inet6 fe80::861f:5390:b5db:3a74%utun1 prefixlen 64 scopeid 0xc
        nd6 options=201<PERFORMNUD,DAD>
```



1. The IP Address of my computer is 192.168.1.102.

2. The value in the upper layer protocol field is ICMP.

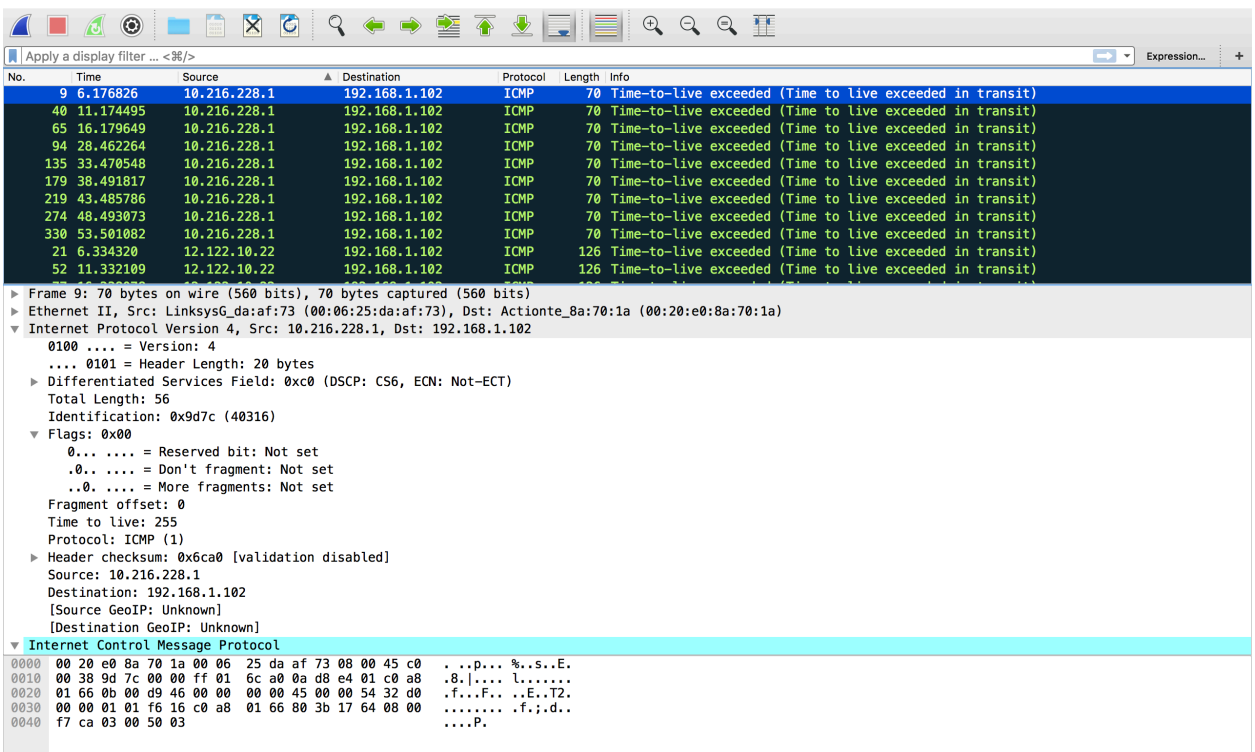
3. There are 20 bytes in the header and a total length of 84 bytes, so the payload of the IP data gram is 64 bytes.

4. The more fragments bit has a value of 0 so thus the data is not fragmented.

5. Identification, Time to live and Header checksum always change between datagrams.

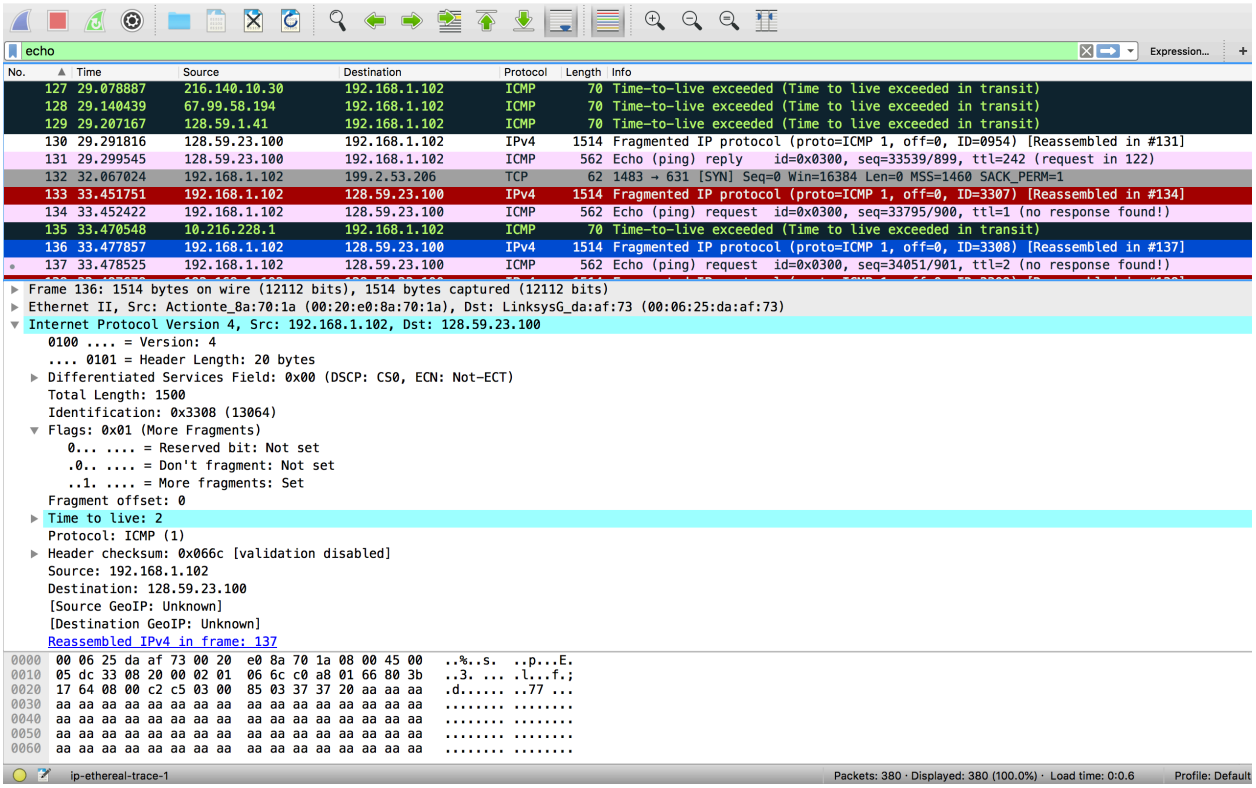
6. The fields that stay constant across IP datagrams are Version, header length, source IP, destination IP, differentiated services, and upper layer protocol. The fields that must stay constant are Version, header length, source ip, destination ip, differentiated services, upper layer protocol. The fields that must change are identification, time to live, and header checksum.

7. The pattern is the IP header identification fields are incremented with the ICMP echo ping request.



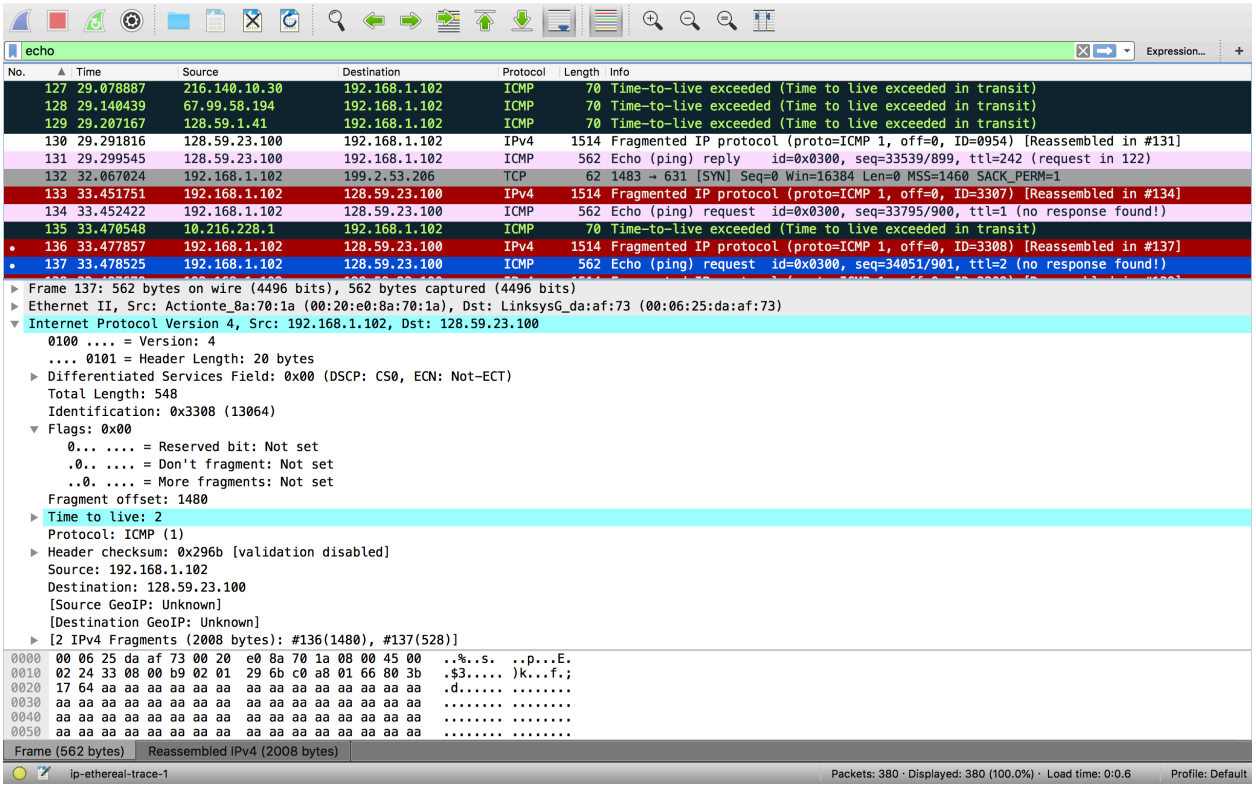
8. The Identification field has 40316 and the TTL is 255.

9. The identification fields changes since it is supposed to be a unique id value. Two or more datagrams having the same id field means that they are part of the same datagram. The TTL field is the same since the TTL for the first hop router is always the same.



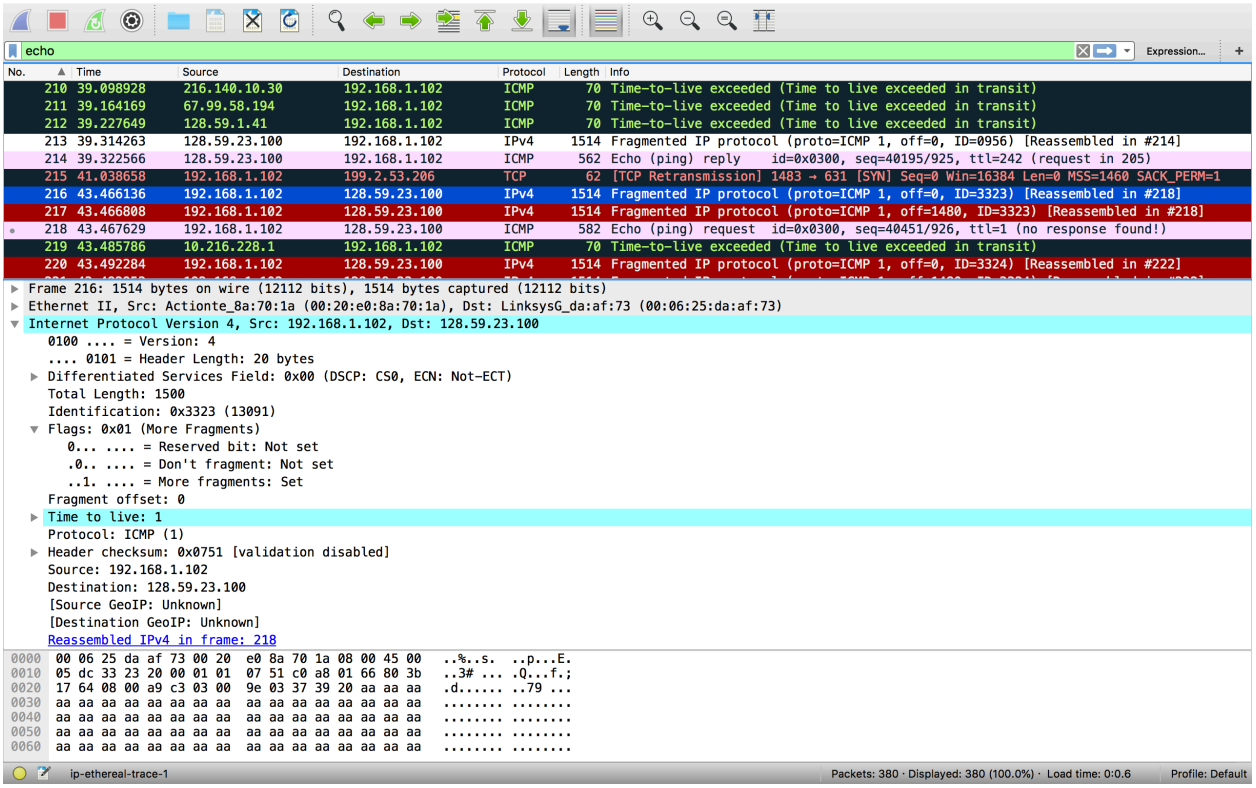
10. This packet was fragmented across more than one ip datagram.

11. Under the flags section, the more fragments bit is set to 1. The fragment offset being 0 denotes that this is the first fragment. The first datagram has a total length of 1500.



12. It is clear that this is the second fragment from looking at the fragment offset which is 1480 here. We know this is the last fragment since the more fragments flag is not set to 1.

13. The fields that changed are total length, flags, fragment offset, and checksum.



14. Switching the packet size to 3500 yielded 3 packets from the original datagram.

15. The fields that changed between all packets are fragment offset and checksum. In the first two packets and the last packets there's a change in total length and the flags that are set.