BARTU KILIÇKAYA

2380640

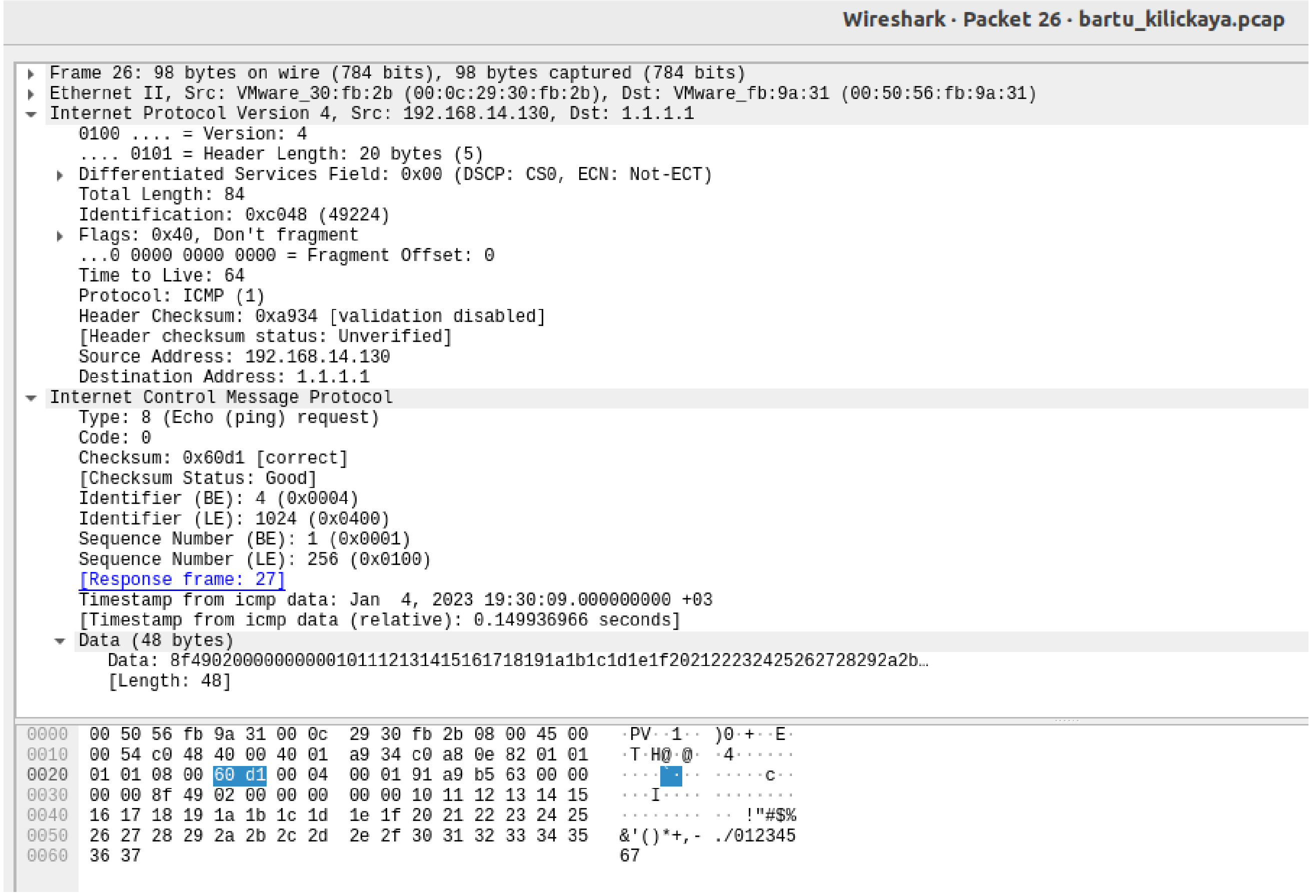
CENG435 COMPUTER NETWORKING

THE4

2 ICMP Packet Analysis

Internet Control Message Protocol” details;

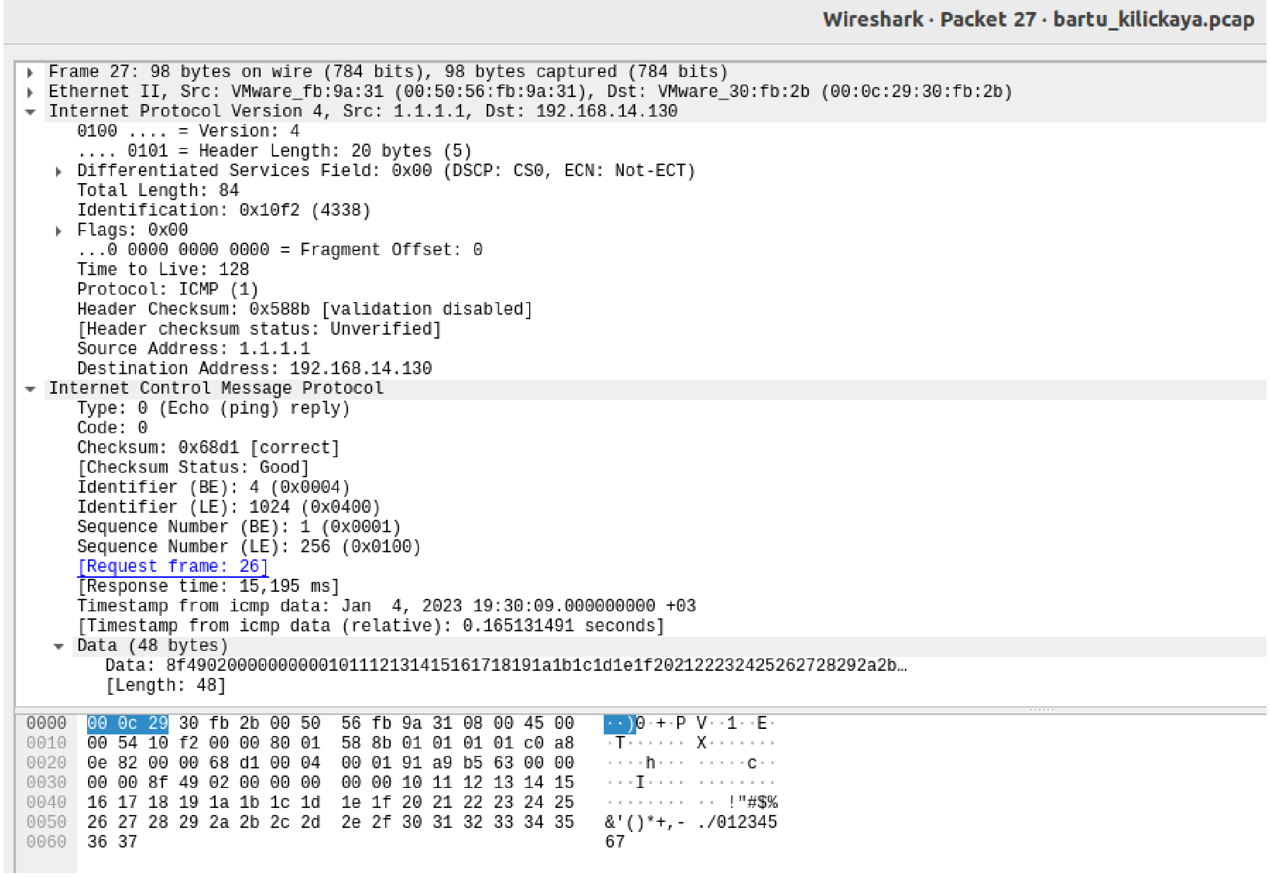
ICMP Request Packet:



Wireshark · Packet 26 · bartu_kilickaya.pcap

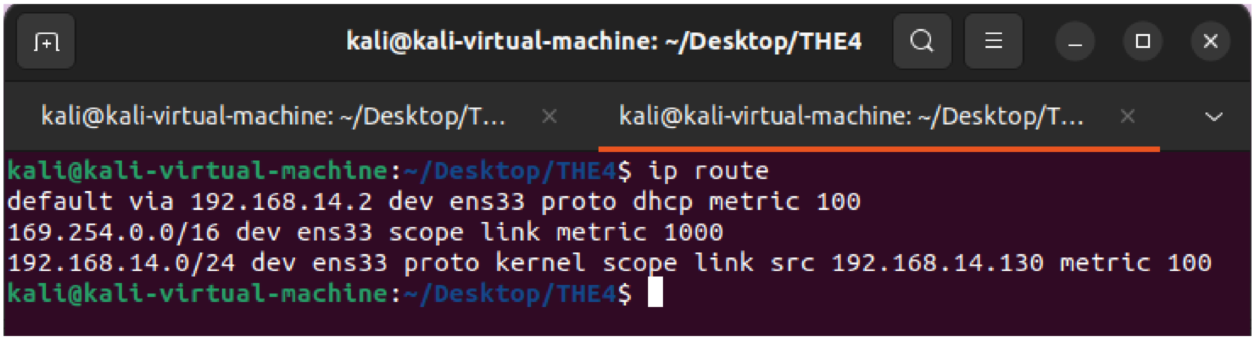
```
▸ Frame 26: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
▸ Ethernet II, Src: VMware_30:fb:2b (00:0c:29:30:fb:2b), Dst: VMware_fb:9a:31 (00:50:56:fb:9a:31)
▾ Internet Protocol Version 4, Src: 192.168.14.130, Dst: 1.1.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 84
    Identification: 0xc048 (49224)
  ▸ Flags: 0x40, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: ICMP (1)
    Header Checksum: 0xa934 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.14.130
    Destination Address: 1.1.1.1
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x60d1 [correct]
    [Checksum Status: Good]
    Identifier (BE): 4 (0x0004)
    Identifier (LE): 1024 (0x0400)
    Sequence Number (BE): 1 (0x0001)
    Sequence Number (LE): 256 (0x0100)
    [Response frame: 27]
    Timestamp from icmp data: Jan  4, 2023 19:30:09.000000000 +03
    [Timestamp from icmp data (relative): 0.149936966 seconds]
  ▾ Data (48 bytes)
      Data: 8f490200000000001011121314151617181919a1b1c1d1e1f202122232425262728292a2b…
      [Length: 48]

0000  00 50 56 fb 9a 31 00 0c  29 30 fb 2b 08 00 45 00   ·PV··1··  )0·+··E·
0010  00 54 c0 48 40 00 40 01  a9 34 c0 a8 0e 82 01 01   ·T·H@·@·  ·4······
0020  01 01 08 00 60 d1 00 04  00 01 91 a9 b5 63 00 00   ····`···  ·····c··
0030  00 00 8f 49 02 00 00 00  00 00 10 11 12 13 14 15   ···I····  ········
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20 21 22 23 24 25   ········  ·· !"#$%
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30 31 32 33 34 35   &'()*+,-  ./012345
0060  36 37                                              67
```

ICMP Response Packet:



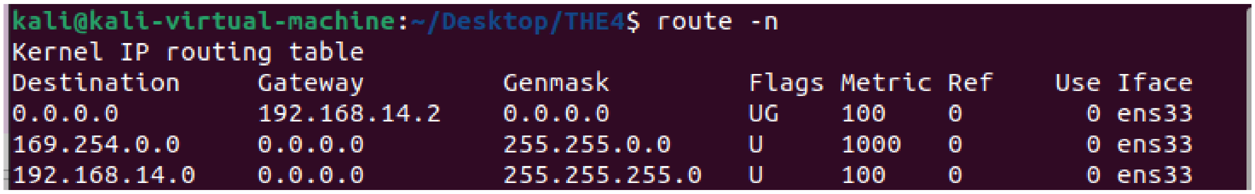
```
▸ Frame 27: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
▸ Ethernet II, Src: VMware_fb:9a:31 (00:50:56:fb:9a:31), Dst: VMware_30:fb:2b (00:0c:29:30:fb:2b)
▾ Internet Protocol Version 4, Src: 1.1.1.1, Dst: 192.168.14.130
     0100 .... = Version: 4
     .... 0101 = Header Length: 20 bytes (5)
   ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
     Total Length: 84
     Identification: 0x10f2 (4338)
   ▸ Flags: 0x00
     ...0 0000 0000 0000 = Fragment Offset: 0
     Time to Live: 128
     Protocol: ICMP (1)
     Header Checksum: 0x588b [validation disabled]
     [Header checksum status: Unverified]
     Source Address: 1.1.1.1
     Destination Address: 192.168.14.130
▾ Internet Control Message Protocol
     Type: 0 (Echo (ping) reply)
     Code: 0
     Checksum: 0x68d1 [correct]
     [Checksum Status: Good]
     Identifier (BE): 4 (0x0004)
     Identifier (LE): 1024 (0x0400)
     Sequence Number (BE): 1 (0x0001)
     Sequence Number (LE): 256 (0x0100)
     [Request frame: 26]
     [Response time: 15,195 ms]
     Timestamp from icmp data: Jan  4, 2023 19:30:09.000000000 +03
     [Timestamp from icmp data (relative): 0.165131491 seconds]
   ▾ Data (48 bytes)
        Data: 8f49020000000000101112131415161718191a1b1c1d1e1f202122232425262728292a2b…
        [Length: 48]

0000  00 0c 29 30 fb 2b 00 50  56 fb 9a 31 08 00 45 00   ··)0·+·P V··1··E·
0010  00 54 10 f2 00 00 80 01  58 8b 01 01 01 01 c0 a8   ·T······ X·······
0020  0e 82 00 00 68 d1 00 04  00 01 91 a9 b5 63 00 00   ····h··· ·····c··
0030  00 00 8f 49 02 00 00 00  00 00 10 11 12 13 14 15   ···I···· ········
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20 21 22 23 24 25   ········ ·· !"#$%
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30 31 32 33 34 35   &'()*+,- ./012345
0060  36 37                                              67
```

Routing table information:



kali@kali-virtual-machine: ~/Desktop/THE4
kali@kali-virtual-machine: ~/Desktop/T...
kali@kali-virtual-machine: ~/Desktop/T...

```
kali@kali-virtual-machine:~/Desktop/THE4$ ip route
default via 192.168.14.2 dev ens33 proto dhcp metric 100
169.254.0.0/16 dev ens33 scope link metric 1000
192.168.14.0/24 dev ens33 proto kernel scope link src 192.168.14.130 metric 100
kali@kali-virtual-machine:~/Desktop/THE4$ 
```



```
kali@kali-virtual-machine:~/Desktop/THE4$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.14.2    0.0.0.0         UG    100    0        0 ens33
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 ens33
192.168.14.0    0.0.0.0         255.255.255.0   U     100    0        0 ens33
```

3 Questions:

1) For ICMP request, the source is 192.168.14.130 and the destination is 1.1.1.1. For ICMP reply, the source is 1.1.1.1 and the destination is 192.168.14.130.

2) No, there is no port number information in both request and reply packets. Because Internet Control Message Protocol is a network-layer protocol and it was designed to communicate network layer information between routers and hosts. IMCP Packet header has “Type” and “Code”. These two fields together define the specific message received. Since the network software uses all ICMP messages, no port numbers are required to direct the ICMP message to an application layer process.

3)

a- The 4-byte ICMP header contains an 8-bit type field, which defines the ICMP type. The type field specifies that why the ICMP packet is used for. For instance 0 for echo reply and 3 for destination unreachable.

b- Depending on the type, the 8-bit code field may also be used, which contains additional information. The code info defines status of the packages. For instance:

Type 3 and code 1: Host is unreachable

Type 3 and code 2: Protocol is unreachable

If the type does not have any codes defined, the code field is set to zero.

c- For request packet, we have type 8 and code 0. Type is set to 8 because it is a echo packet and code is set to 0 since for type 8, there aren’t any codes defined.

For reply packet, we have type 0 and code 0. Type is set to 0 because it is a echo reply packet and code is set to 0 since for type 0, there aren’t any codes defined.

4)

98 bytes are transferred in total. 14 bytes for Ethernet header, 20 bytes for IP header, 8 bytes for ICMP header and 56 bytes for the ICMP payload.

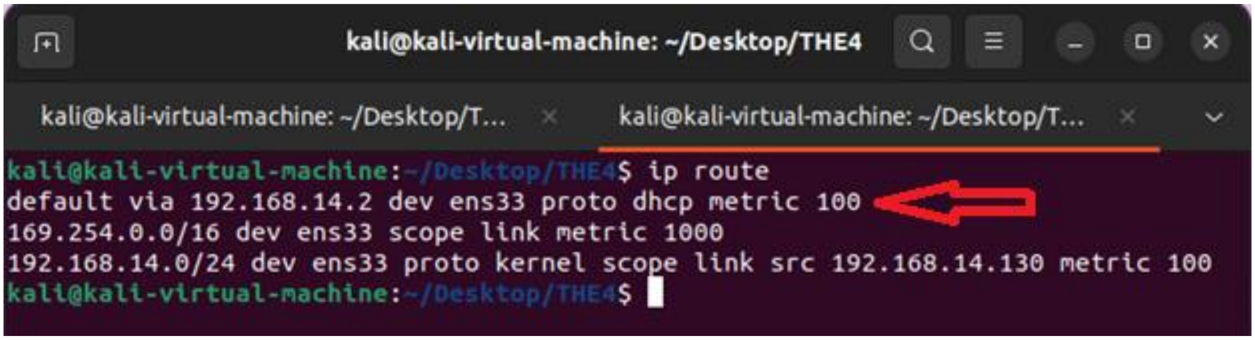
Ethernet header contains destination(6bytes) and source(6 bytes) mac addresses and type field(2 bytes) for internet protocol version (ipv4 or ipv6)

IP header contains version(1), differentiated service field(1), total length(2), flags(1),identification(2), TTL(1), protocol(1), header checksum(2), source and destination address(4 bytes each). Since this is a request packet, source IP is our IP and destination IP is 1.1.1.1. Protocol is IMCP as expected. Header checksum is used for error detection and TTL is a field that an IP packet represents the maximum number of IP routers that the packet can go through before being discarded.

ICMP header has type(1), code(1) and checksum(2),identifier(2) and sequence number(2) fields. Type specifies that why ICMP packet is used for, code defines the status of the packages. Checksum is for error detection and The Identifier and Sequence Number may be used by the echo sender to aid in matching the replies with the echo requests.

56 byte IMCP payload contains 48 byte data and 8 byte timestamp. Timestamp is used by network routers to synchronize their system clocks for time and date.

5)

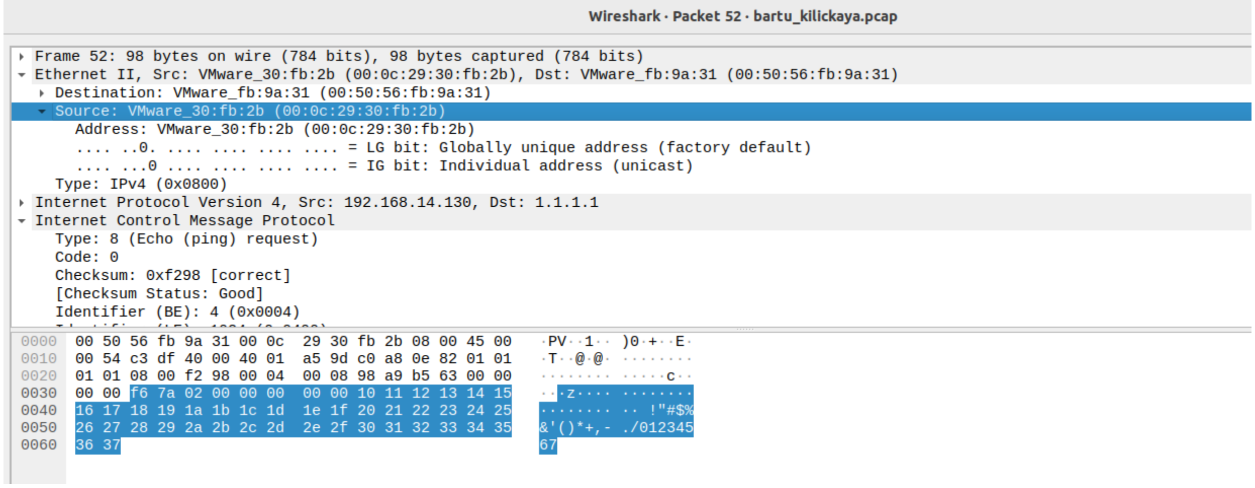

kali@kali-virtual-machine: ~/Desktop/THE4
kali@kali-virtual-machine: ~/Desktop/T...
kali@kali-virtual-machine: ~/Desktop/T...

```
kali@kali-virtual-machine:~/Desktop/THE4$ ip route
default via 192.168.14.2 dev ens33 proto dhcp metric 100
169.254.0.0/16 dev ens33 scope link metric 1000
192.168.14.0/24 dev ens33 proto kernel scope link src 192.168.14.130 metric 100
kali@kali-virtual-machine:~/Desktop/THE4$
```

We should remove the default one which corresponds to destination: 0.0.0.0 , gateway: 192.168.14.2 , genmask: 0.0.0.0 . Since 1.1.1.1 does not belong to neither 169.254.0.0 nor 192.168.14.0 networks, removing these don't inhibit pinging to 1.1.1.1. However since default destination is 0.0.0.0 and corresponding genmask is 0.0.0.0, 1.1.1.1 is in this default route and removing this causes network is unreachable error as expected.

6)

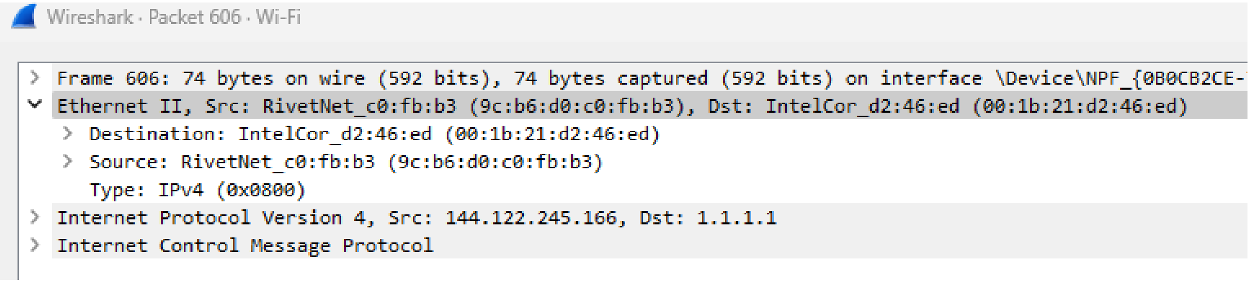
a- Since I'm using Vmware for doing this homework in linux machine, I see 00:0c:29:30:fb:2b. However this is not my correct 48 bit ethernet address of my machine since I'm using virtual machine. If I hadn't used virtual machine, I would have seen my normal 48 bit ethernet address.



Wireshark · Packet 52 · bartu_kilickaya.pcap

```
› Frame 52: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
⌄ Ethernet II, Src: VMware_30:fb:2b (00:0c:29:30:fb:2b), Dst: VMware_fb:9a:31 (00:50:56:fb:9a:31)
  › Destination: VMware_fb:9a:31 (00:50:56:fb:9a:31)
  ⌄ Source: VMware_30:fb:2b (00:0c:29:30:fb:2b)
      Address: VMware_30:fb:2b (00:0c:29:30:fb:2b)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
› Internet Protocol Version 4, Src: 192.168.14.130, Dst: 1.1.1.1
⌄ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xf298 [correct]
    [Checksum Status: Good]
    Identifier (BE): 4 (0x0004)

0000  00 50 56 fb 9a 31 00 0c  29 30 fb 2b 08 00 45 00   ·PV··1·· )0·+··E·
0010  00 54 c3 df 40 00 40 01  a5 9d c0 a8 0e 82 01 01   ·T··@·@· ········
0020  01 01 08 00 f2 98 00 04  00 08 98 a9 b5 63 00 00   ········ ·····c··
0030  00 00 f6 7a 02 00 00 00  00 00 10 11 12 13 14 15   ···z···· ········
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20 21 22 23 24 25   ········ ·· !"#$%
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30 31 32 33 34 35   &'()*+,- ./012345
0060  36 37                                              67
```

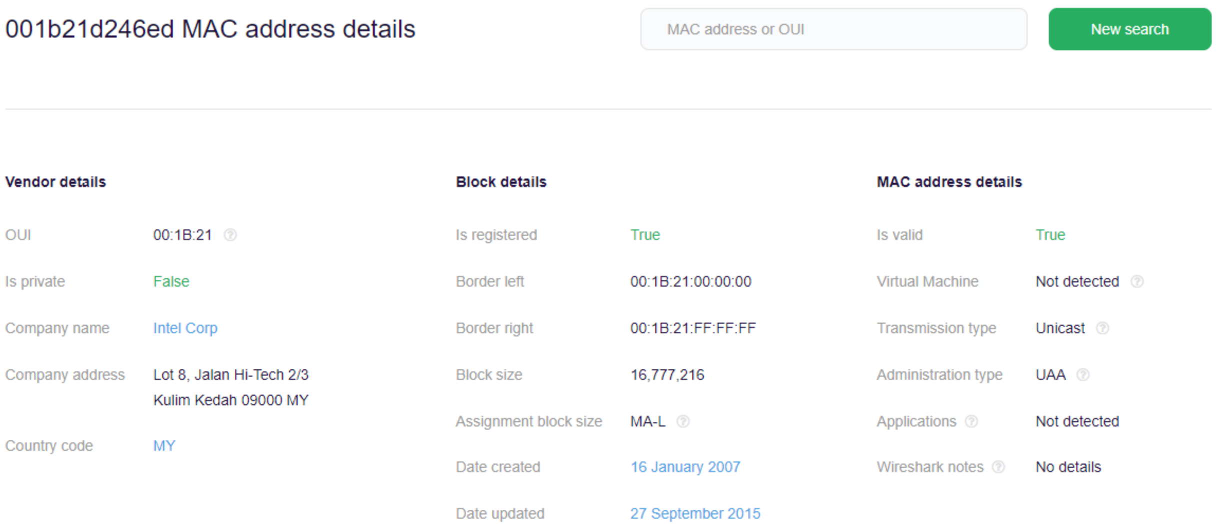
Windows capture just to see my 48 bit ethernet address:



Wireshark · Packet 606 · Wi-Fi

```
› Frame 606: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{0B0CB2CE-
⌄ Ethernet II, Src: RivetNet_c0:fb:b3 (9c:b6:d0:c0:fb:b3), Dst: IntelCor_d2:46:ed (00:1b:21:d2:46:ed)
  › Destination: IntelCor_d2:46:ed (00:1b:21:d2:46:ed)
  › Source: RivetNet_c0:fb:b3 (9c:b6:d0:c0:fb:b3)
    Type: IPv4 (0x0800)
› Internet Protocol Version 4, Src: 144.122.245.166, Dst: 1.1.1.1
› Internet Control Message Protocol
```

b- Destination address is 00:50:56:fb:9a:31 . If we lookup this address, as expected it belongs to Vmware since I'm using virtual machine. However; if I check the correct destination address which is shown on the second screenshot, the destination address is 00:1b:21:d2:46:ed . If we lookup this address, this machine belongs to Intel Corp:



001b21d246ed MAC address details

Vendor details

OUI	00:1B:21
Is private	False
Company name	Intel Corp
Company address	Lot 8, Jalan Hi-Tech 2/3 Kulim Kedah 09000 MY
Country code	MY

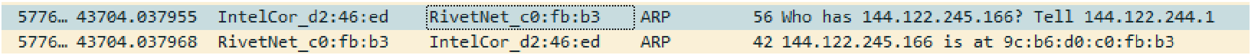
Block details

Is registered	True
Border left	00:1B:21:00:00:00
Border right	00:1B:21:FF:FF:FF
Block size	16,777,216
Assignment block size	MA-L
Date created	16 January 2007
Date updated	27 September 2015

MAC address details

Is valid	True
Virtual Machine	Not detected
Transmission type	Unicast
Administration type	UAA
Applications	Not detected
Wireshark notes	No details

c- In all 99 packets, I have only seen Iptv4 in type field, however we could have seen ARP,Ipv6,AppleTalk,etc. For instance we could have seen something like this:



```
5776… 43704.037955  IntelCor_d2:46:ed   RivetNet_c0:fb:b3  ARP      56 Who has 144.122.245.166? Tell 144.122.244.1
5776… 43704.037968  RivetNet_c0:fb:b3   IntelCor_d2:46:ed  ARP      42 144.122.245.166 is at 9c:b6:d0:c0:fb:b3
```

The Address Resolution Protocol (ARP) is a communication protocol used for discovering the link layer address, such as a MAC address, associated with a given internet layer address, typically an IPv4 address. In this example, our MAC address is asked in that network using our Ipv4 address. Our computer returns our MAC address since asked IP belongs to us.

Internet Protocol version 4 (IPv4) is used in all of my packets since it defines how the addressing works and how network hosts can be identified and found on the network.