Middle East Technical University Department of Computer Engineering

CENG 435

Data Communications and Networking

Fall '2021-2022

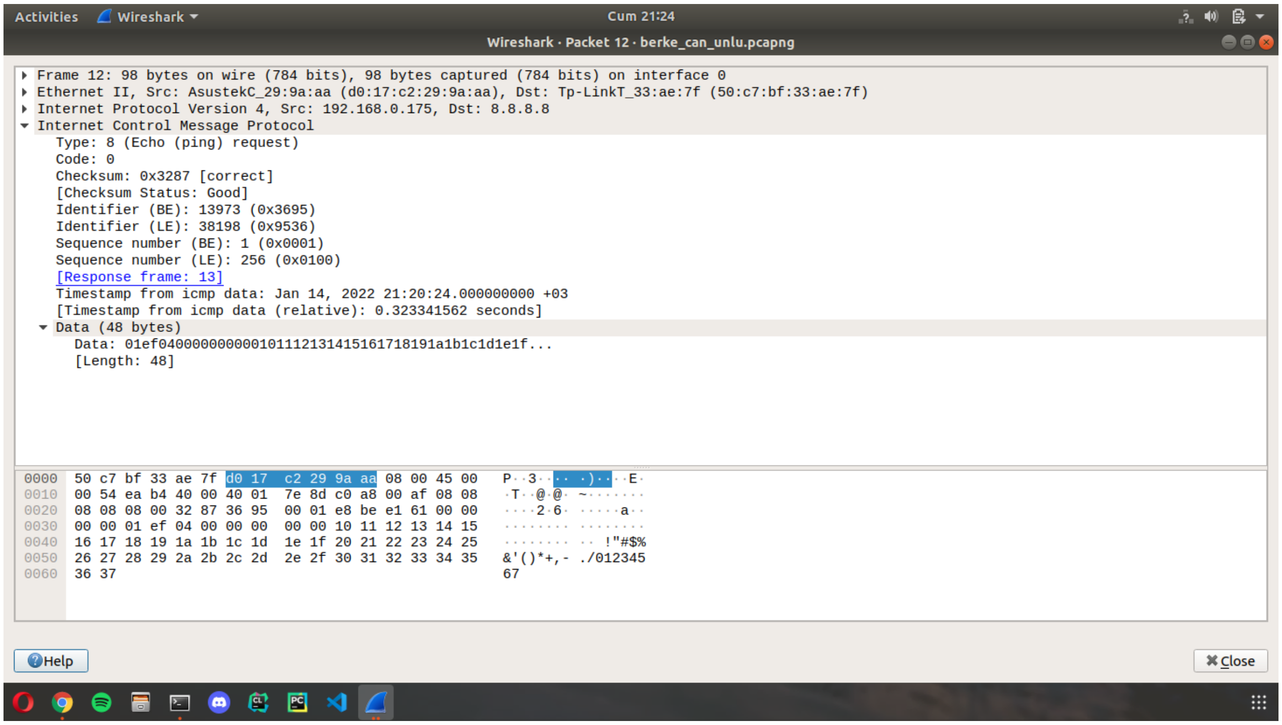
Homework 4

Student Name and Surname: Berke Can Ünlü
Student Number: 2381028

1 ICMP Packet Analysis

1.1 Screenshots of ICMP Request, Reply, and Routing Table

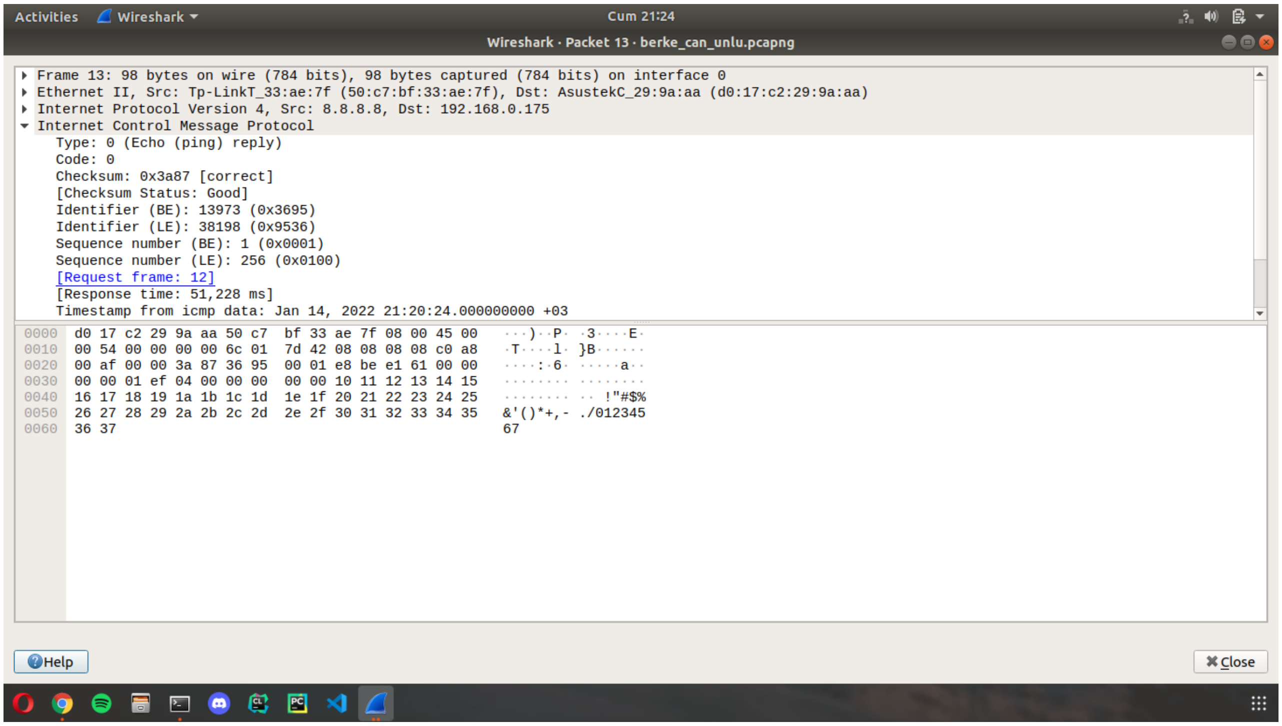
1.1.1 ICMP Request



```
▸ Frame 12: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa), Dst: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f)
▸ Internet Protocol Version 4, Src: 192.168.0.175, Dst: 8.8.8.8
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x3287 [correct]
    [Checksum Status: Good]
    Identifier (BE): 13973 (0x3695)
    Identifier (LE): 38198 (0x9536)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Response frame: 13]
    Timestamp from icmp data: Jan 14, 2022 21:20:24.000000000 +03
    [Timestamp from icmp data (relative): 0.323341562 seconds]
  ▾ Data (48 bytes)
      Data: 01ef0400000000000101112131415161718191a1b1c1d1e1f...
      [Length: 48]
```

```
0000  50 c7 bf 33 ae 7f d0 17  c2 29 9a aa 08 00 45 00   P··3···· ·)····E·
0010  00 54 ea b4 40 00 40 01  7e 8d c0 a8 00 af 08 08   ·T··@·@· ~·······
0020  08 08 08 00 32 87 36 95  00 01 e8 be e1 61 00 00   ····2·6· ·····a··
0030  00 00 01 ef 04 00 00 00  00 00 10 11 12 13 14 15   ········ ········
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
```

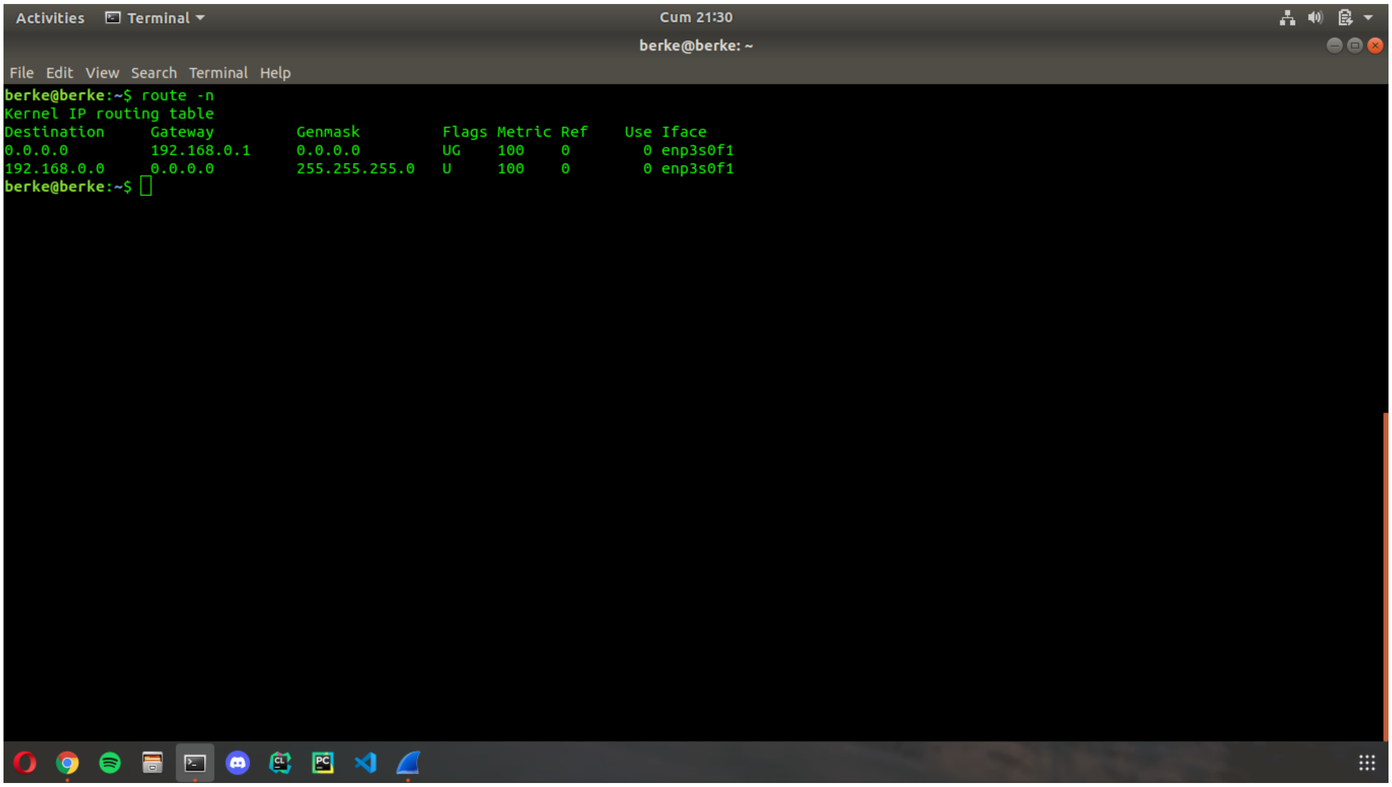
1.1.2 ICMP Reply



```
▸ Frame 13: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f), Dst: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa)
▸ Internet Protocol Version 4, Src: 8.8.8.8, Dst: 192.168.0.175
▾ Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x3a87 [correct]
    [Checksum Status: Good]
    Identifier (BE): 13973 (0x3695)
    Identifier (LE): 38198 (0x9536)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Request frame: 12]
    [Response time: 51,228 ms]
    Timestamp from icmp data: Jan 14, 2022 21:20:24.000000000 +03
```

```
0000  d0 17 c2 29 9a aa 50 c7  bf 33 ae 7f 08 00 45 00   ···)··P· ·3····E·
0010  00 54 00 00 00 00 6c 01  7d 42 08 08 08 08 c0 a8   ·T····l· }B······
0020  00 af 00 00 3a 87 36 95  00 01 e8 be e1 61 00 00   ····:·6· ·····a··
0030  00 00 01 ef 04 00 00 00  00 00 10 11 12 13 14 15   ········ ········
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
```

1.1.3 Routing Table

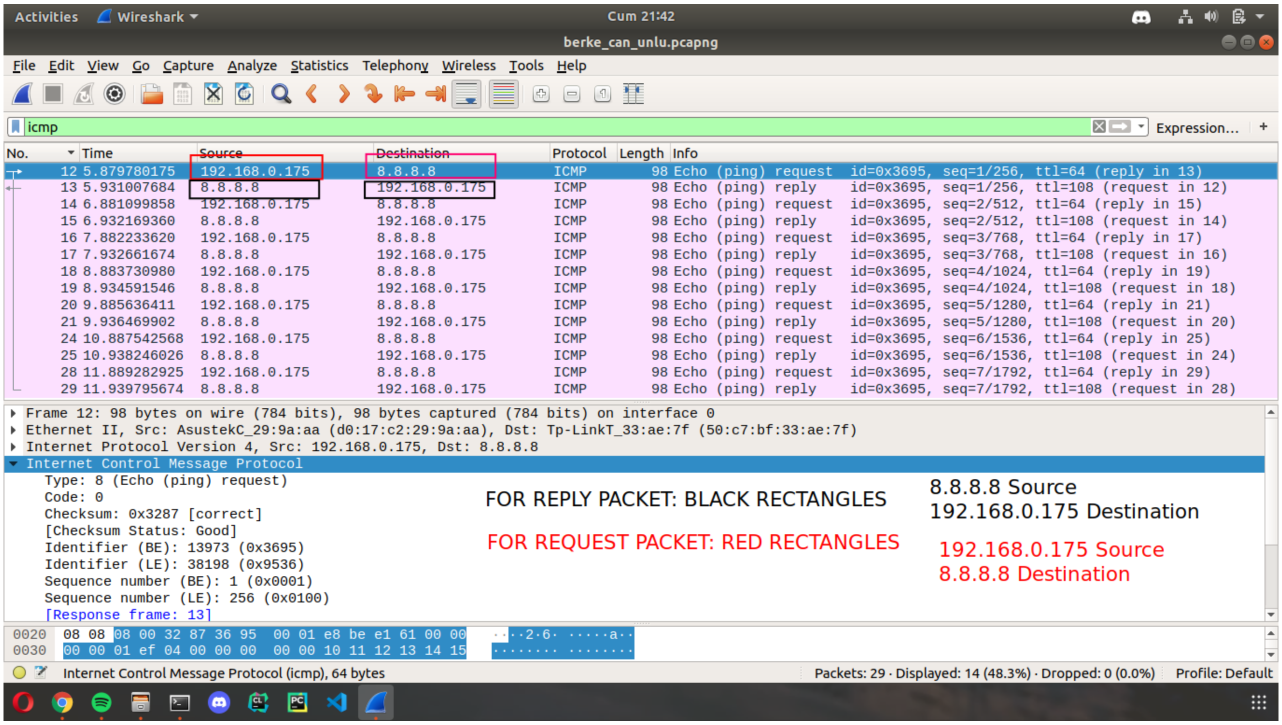


```
berke@berke:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.0.1     0.0.0.0         UG    100    0        0 enp3s0f1
192.168.0.0     0.0.0.0         255.255.255.0   U     100    0        0 enp3s0f1
berke@berke:~$ 
```

2 Answers

2.1 1)

Source address is 192.168.0.175 and Destination address is 8.8.8.8 for request packets.
Source address is 8.8.8.8 and Destination address is 192.168.0.175 for reply packets.


Activities
Wireshark
Cum 21:42
berke_can_unlu.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression…
+

```
No.   Time         Source          Destination     Protocol Length Info
   12 5.879780175  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=1/256, ttl=64 (reply in 13)
   13 5.931007684  8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=1/256, ttl=108 (request in 12)
   14 6.881099858  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=2/512, ttl=64 (reply in 15)
   15 6.932169360  8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=2/512, ttl=108 (request in 14)
   16 7.882233620  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=3/768, ttl=64 (reply in 17)
   17 7.932661674  8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=3/768, ttl=108 (request in 16)
   18 8.883730980  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=4/1024, ttl=64 (reply in 19)
   19 8.934591546  8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=4/1024, ttl=108 (request in 18)
   20 9.885636411  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=5/1280, ttl=64 (reply in 21)
   21 9.936469902  8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=5/1280, ttl=108 (request in 20)
   24 10.887542568 192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=6/1536, ttl=64 (reply in 25)
   25 10.938246026 8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=6/1536, ttl=108 (request in 24)
   28 11.889282925 192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=7/1792, ttl=64 (reply in 29)
   29 11.939795674 8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=7/1792, ttl=108 (request in 28)

▸ Frame 12: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa), Dst: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f)
▸ Internet Protocol Version 4, Src: 192.168.0.175, Dst: 8.8.8.8
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x3287 [correct]
    [Checksum Status: Good]
    Identifier (BE): 13973 (0x3695)
    Identifier (LE): 38198 (0x9536)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Response frame: 13]

0020  08 08 08 00 32 87 36 95  00 01 e8 be e1 61 00 00   ··· ·2·6· ·····a··
0030  00 00 01 ef 04 00 00 00  00 00 10 11 12 13 14 15   ········ ········

Internet Control Message Protocol (icmp), 64 bytes    Packets: 29 · Displayed: 14 (48.3%) · Dropped: 0 (0.0%)    Profile: Default
```

FOR REPLY PACKET: BLACK RECTANGLES

8.8.8.8 Source
192.168.0.175 Destination

FOR REQUEST PACKET: RED RECTANGLES

192.168.0.175 Source
8.8.8.8 Destination

2.2 2)

There is no port number information in both request and reply packets. ICMP packets does not have port number since it was designed in order to communicate network-layer information between hosts and routers. It was not designed to communicate between application layer processes. It uses type and code instead of port number.[1]

[1] https://www.howtouselinux.com/post/icmp-port-number

2.3 3)

2.3.1 a)

ICMP type occupies the first 1 byte of ICMP message header. The purpose of this field is that it provides a brief explanation regarding the message. It gives information about what the message is for. So, the receiving device understands why it is getting message and how to treat it.[2]

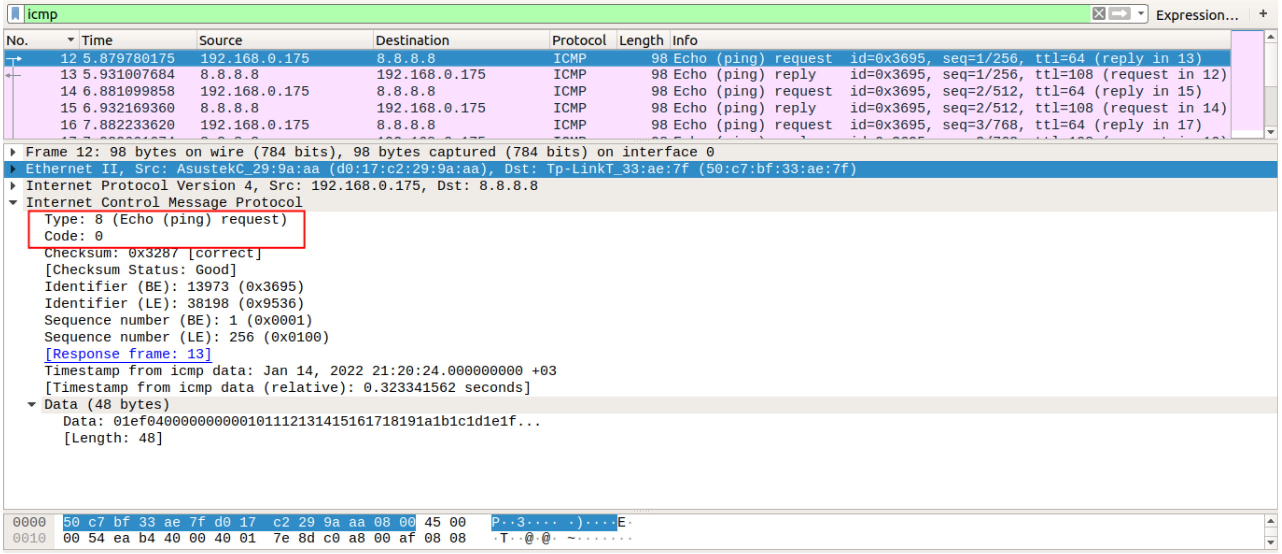
2.3.2 b)

ICMP code occupies the second byte of ICMP message header. The purpose of this field is that it specifies what kind of ICMP message that is sent.[3] Every type field has own code fields.
For example, for Type 11 (Time Exceeded), there is 2 codes such that, 0 for Time to Live exceeded in transit, and 1 for Fragment reassembly time exceeded.[4]

2.3.3 c)

I will explain type and code fields for both request packets and reply packets respectively.

In request packet, Type field equals to 8. This indicates that this message was sent for "Echo (ping) request".[5] Its code is 0 since this type has no code information.



icmp

```
No.   Time          Source          Destination     Protocol Length Info
   12 5.879780175   192.168.0.175   8.8.8.8         ICMP        98 Echo (ping) request  id=0x3695, seq=1/256, ttl=64 (reply in 13)
   13 5.931007684   8.8.8.8         192.168.0.175   ICMP        98 Echo (ping) reply    id=0x3695, seq=1/256, ttl=108 (request in 12)
   14 6.881099858   192.168.0.175   8.8.8.8         ICMP        98 Echo (ping) request  id=0x3695, seq=2/512, ttl=64 (reply in 15)
   15 6.932169360   8.8.8.8         192.168.0.175   ICMP        98 Echo (ping) reply    id=0x3695, seq=2/512, ttl=108 (request in 14)
   16 7.882233620   192.168.0.175   8.8.8.8         ICMP        98 Echo (ping) request  id=0x3695, seq=3/768, ttl=64 (reply in 17)
```

```
▸ Frame 12: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa), Dst: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f)
▸ Internet Protocol Version 4, Src: 192.168.0.175, Dst: 8.8.8.8
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x3287 [correct]
    [Checksum Status: Good]
    Identifier (BE): 13973 (0x3695)
    Identifier (LE): 38198 (0x9536)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Response frame: 13]
    Timestamp from icmp data: Jan 14, 2022 21:20:24.000000000 +03
    [Timestamp from icmp data (relative): 0.323341562 seconds]
  ▾ Data (48 bytes)
      Data: 01ef0400000000000101112131415161718191a1b1c1d1e1f...
      [Length: 48]
```

```
0000  50 c7 bf 33 ae 7f d0 17  c2 29 9a aa 08 00 45 00   P··3···· ·)····E·
0010  00 54 ea b4 40 00 40 01  7e 8d c0 a8 00 af 08 08   ·T··@·@· ~·······
```

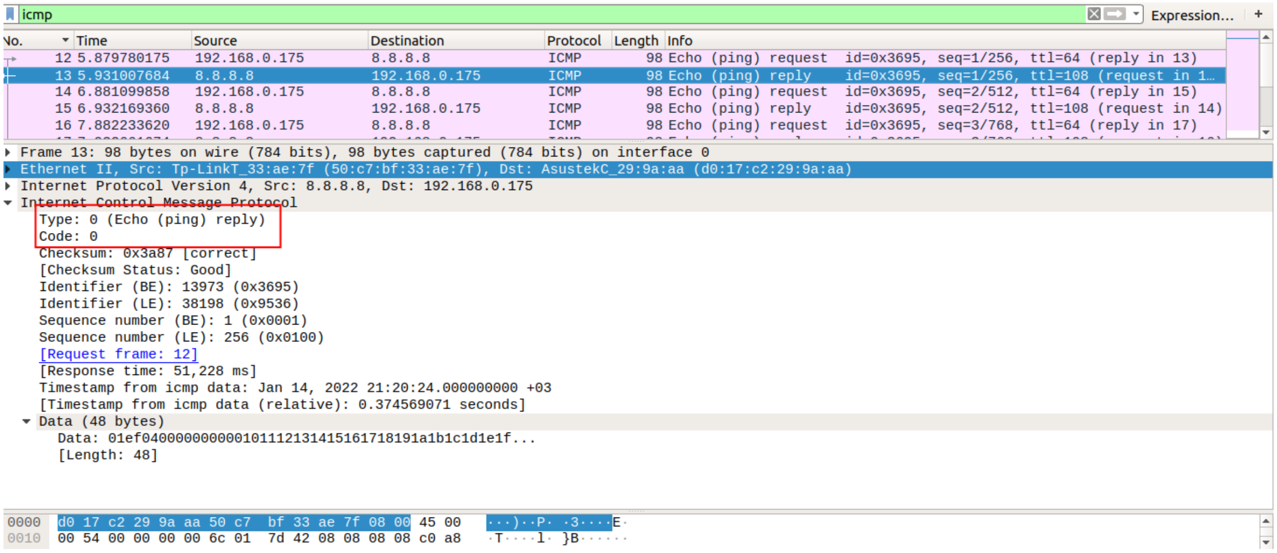
[2] https://www.howtouselinux.com/post/icmp-type

[3] https://networklessons.com/cisco/ccie-routing-switching-written/icmp-internet-control-message-protocol

[4] https://www.ibm.com/docs/en/qsip/7.4?topic=applications-icmp-type-code-ids

[5] https://www.ibm.com/docs/en/qsip/7.4?topic=applications-icmp-type-code-ids

In reply packet, Type field equals to 0. This indicates that this message was sent for ”Echo (ping) reply”.[6] Its code is 0 since this type has no code information.



```
No.    Time         Source          Destination     Protocol Length Info
  12 5.879780175  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=1/256, ttl=64 (reply in 13)
  13 5.931007684  8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=1/256, ttl=108 (request in 1…
  14 6.881099858  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=2/512, ttl=64 (reply in 15)
  15 6.932169360  8.8.8.8         192.168.0.175   ICMP       98 Echo (ping) reply    id=0x3695, seq=2/512, ttl=108 (request in 14)
  16 7.882233620  192.168.0.175   8.8.8.8         ICMP       98 Echo (ping) request  id=0x3695, seq=3/768, ttl=64 (reply in 17)
```

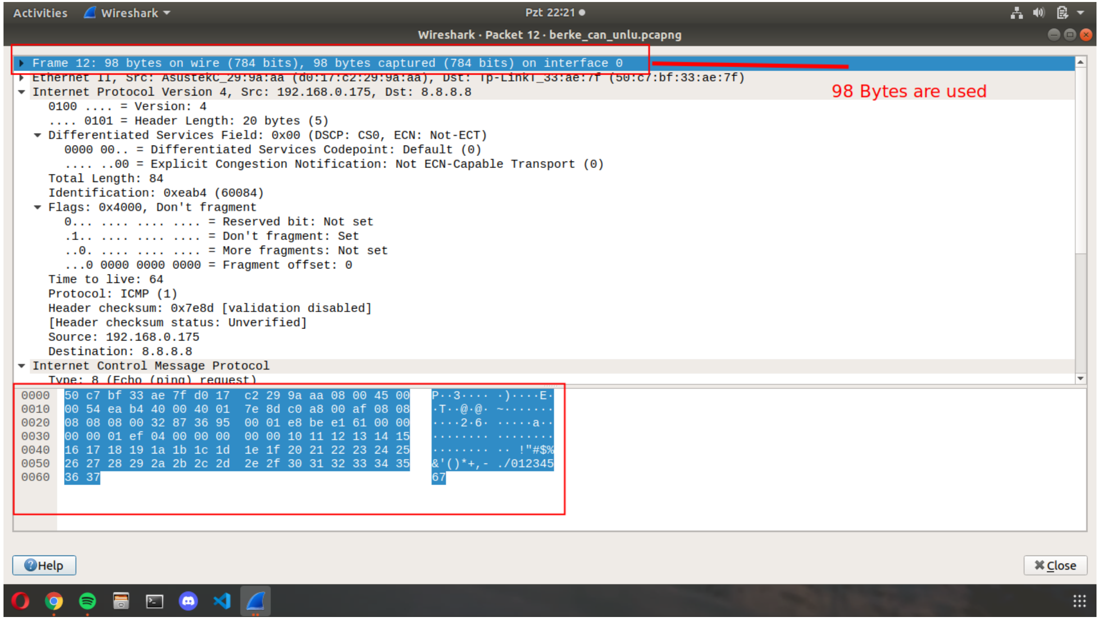
```
▸ Frame 13: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f), Dst: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa)
▸ Internet Protocol Version 4, Src: 8.8.8.8, Dst: 192.168.0.175
▾ Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x3a87 [correct]
    [Checksum Status: Good]
    Identifier (BE): 13973 (0x3695)
    Identifier (LE): 38198 (0x9536)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Request frame: 12]
    [Response time: 51,228 ms]
    Timestamp from icmp data: Jan 14, 2022 21:20:24.000000000 +03
    [Timestamp from icmp data (relative): 0.374569071 seconds]
  ▾ Data (48 bytes)
      Data: 01ef0400000000000101112131415161718191a1b1c1d1e1f...
      [Length: 48]

0000  d0 17 c2 29 9a aa 50 c7  bf 33 ae 7f 08 00 45 00   ···)··P· ·3····E·
0010  00 54 00 00 00 00 6c 01  7d 42 08 08 08 08 c0 a8   ·T····l· }B······
```

Echo Request and Echo reply are used to test destination accessibility and status. A host sends an Echo Request and listens for a corresponding Echo Reply. This is most commonly done using the ping command.[7]

2.4 4)

The total number of bytes that were sent is 98.



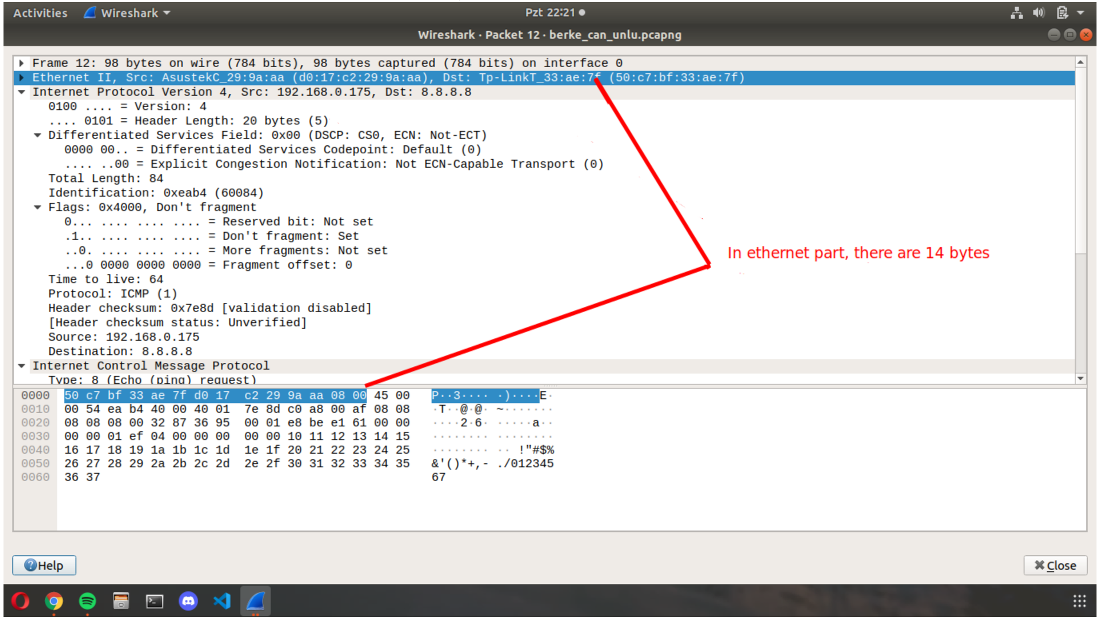
```
▸ Frame 12: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa), Dst: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f)
▾ Internet Protocol Version 4, Src: 192.168.0.175, Dst: 8.8.8.8
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▾ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 84
    Identification: 0xeab4 (60084)
  ▾ Flags: 0x4000, Don't fragment
      0... .... .... .... = Reserved bit: Not set
      .1.. .... .... .... = Don't fragment: Set
      ..0. .... .... .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 64
    Protocol: ICMP (1)
    Header checksum: 0x7e8d [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.0.175
    Destination: 8.8.8.8
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)

0000  50 c7 bf 33 ae 7f d0 17  c2 29 9a aa 08 00 45 00   P··3···· ·)····E·
0010  00 54 ea b4 40 00 40 01  7e 8d c0 a8 00 af 08 08   ·T··@·@· ~·······
0020  08 08 08 00 32 87 36 95  00 01 e8 be e1 61 00 00   ····2·6· ·····a··
0030  00 00 01 ef 04 00 00 00  00 00 10 11 12 13 14 15   ········ ········
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
```

[6] https://www.ibm.com/docs/en/qsip/7.4?topic=applications-icmp-type-code-ids

[7] https://docs.sophos.com/esg/enterprise-console/5-5/help/en-us/esg/Enterprise-Console/concepts/Further$_information_o$

In Ethernet part, there are 14 bytes.

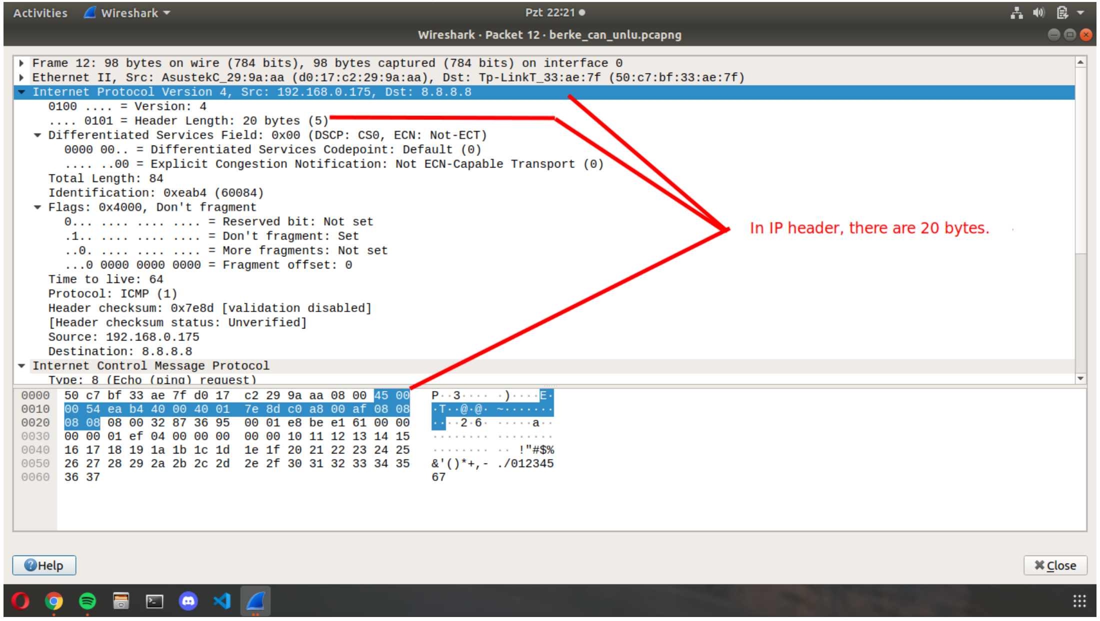


```
▸ Frame 12: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa), Dst: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f)
▾ Internet Protocol Version 4, Src: 192.168.0.175, Dst: 8.8.8.8
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▾ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 84
    Identification: 0xeab4 (60084)
  ▾ Flags: 0x4000, Don't fragment
      0... .... .... .... = Reserved bit: Not set
      .1.. .... .... .... = Don't fragment: Set
      ..0. .... .... .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 64
    Protocol: ICMP (1)
    Header checksum: 0x7e8d [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.0.175
    Destination: 8.8.8.8
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)

0000  50 c7 bf 33 ae 7f d0 17  c2 29 9a aa 08 00 45 00   P··3····  ·)····E·
0010  00 54 ea b4 40 00 40 01  7e 8d c0 a8 00 af 08 08   ·T··@·@·  ~·······
0020  08 08 08 00 32 87 36 95  00 01 e8 be e1 61 00 00   ····2·6·  ·····a··
0030  00 00 01 ef 04 00 00 00  00 00 10 11 12 13 14 15   ········  ········
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
```

In ethernet part, there are 14 bytes

In IP header, the packet has 20 bytes.



```
▸ Frame 12: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: AsustekC_29:9a:aa (d0:17:c2:29:9a:aa), Dst: Tp-LinkT_33:ae:7f (50:c7:bf:33:ae:7f)
▾ Internet Protocol Version 4, Src: 192.168.0.175, Dst: 8.8.8.8
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▾ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 84
    Identification: 0xeab4 (60084)
  ▾ Flags: 0x4000, Don't fragment
      0... .... .... .... = Reserved bit: Not set
      .1.. .... .... .... = Don't fragment: Set
      ..0. .... .... .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 64
    Protocol: ICMP (1)
    Header checksum: 0x7e8d [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.0.175
    Destination: 8.8.8.8
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)

0000  50 c7 bf 33 ae 7f d0 17  c2 29 9a aa 08 00 45 00   P··3····  ·)····E·
0010  00 54 ea b4 40 00 40 01  7e 8d c0 a8 00 af 08 08   ·T··@·@·  ~·······
0020  08 08 08 00 32 87 36 95  00 01 e8 be e1 61 00 00   ····2·6·  ·····a··
0030  00 00 01 ef 04 00 00 00  00 00 10 11 12 13 14 15   ········  ········
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
```

In IP header, there are 20 bytes.

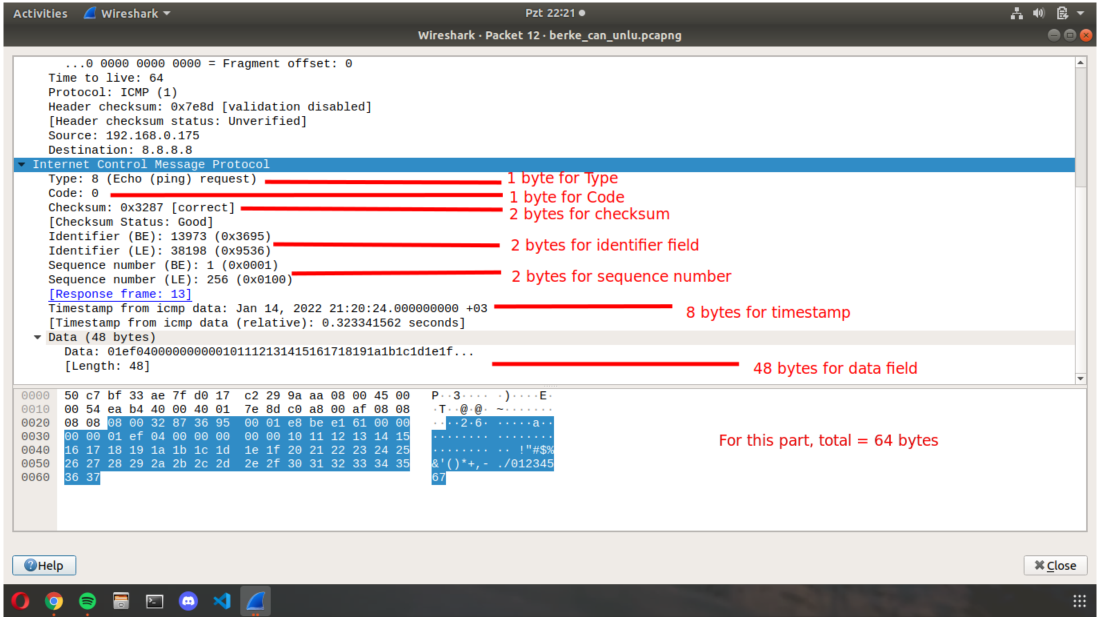
The packet has 1 byte for TYPE, 1 byte for CODE part.

Checksum occupies 2 bytes of the packet.

Identifier field has 2 bytes. Also, sequence number field has 2 bytes as well. Identifier field matches Echo request with Echo reply. Sequence number field increments by one for each Echo request sent. These two numbers are sent back to the Echo issuer in the Echo Reply.[8] In other words, they are sent in order to help matching the replies with requests.[9]

Timestamp from ICMP data occupies 8 bytes of the packet.

ICMP header is 16 bytes. This header includes: Type, Code, Identifier, Sequence number and timestamp.



```
      ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 64
    Protocol: ICMP (1)
    Header checksum: 0x7e8d [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.0.175
    Destination: 8.8.8.8
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x3287 [correct]
    [Checksum Status: Good]
    Identifier (BE): 13973 (0x3695)
    Identifier (LE): 38198 (0x9536)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Response frame: 13]
    Timestamp from icmp data: Jan 14, 2022 21:20:24.000000000 +03
    [Timestamp from icmp data (relative): 0.323341562 seconds]
  ▾ Data (48 bytes)
      Data: 01ef0400000000001011121314151617181 91a1b1c1d1e1f...
      [Length: 48]
```

```
0000  50 c7 bf 33 ae 7f d0 17  c2 29 9a aa 08 00 45 00   P··3····  ·)····E·
0010  00 54 ea b4 40 00 40 01  7e 8d c0 a8 00 af 08 08   ·T··@·@·  ~·······
0020  08 08 08 00 32 87 36 95  00 01 e8 be e1 61 00 00   ····2·6·  ·····a··
0030  00 00 01 ef 04 00 00 00  00 00 10 11 12 13 14 15   ········  ········
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
```

1 byte for Type
1 byte for Code
2 bytes for checksum
2 bytes for identifier field
2 bytes for sequence number
8 bytes for timestamp
48 bytes for data field
For this part, total = 64 bytes

And the rest of the packet includes data field. In data field, 48 bytes are used.

Eventually, $Total = Ethernet + IP\ Header + ICMP\ Packet = 14 + 20 + 64 = 98$ bytes.

2.5 5)

I should remove the first rule. The Destination column identifies the destination network. The Gateway column identifies the defined gateway for the specified network. The Genmask column shows the netmask for the network.[10] 0.0.0.0 is the default destination. If the destination address is not specified in the routing table, packets will follow this rule since it is default.[11] This rule indicates that send packet to destination address "0.0.0.0" with using gateway 192.168.0.1 and genmask "0.0.0.0" . If I remove this rule, my machine cannot send any ping request since packets will drop.

In second line, destination is the Local Network since the destination address is "192.168.0.0".

[8] https://www.rhyshaden.com/icmp.htm

[9] http://www.networksorcery.com/enp/protocol/icmp/msg8.htm

[10] https://www.techrepublic.com/article/understand-the-basics-of-linux-routing/

[11] https://opensource.com/business/16/8/introduction-linux-network-routing