

ELEC 331 Assignment 3

Wireshark IP

1	0.000000	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=103/26368, ttl=255 (reply in 9)
2	0.021406	35.186.224.53	192.168.1.64	TLSv1...	117 Application Data
3	0.021697	192.168.1.64	35.186.224.53	TCP	54 54713 → 443 [FIN, ACK] Seq=1 Ack=64 Win=257 Len=0
4	0.022318	35.186.224.53	192.168.1.64	TCP	60 443 → 54713 [FIN, ACK] Seq=64 Ack=1 Win=254 Len=0
5	0.022344	192.168.1.64	35.186.224.53	TCP	54 54713 → 443 [ACK] Seq=2 Ack=65 Win=257 Len=0
6	0.031146	35.186.224.53	192.168.1.64	TCP	60 443 → 54713 [ACK] Seq=65 Ack=2 Win=254 Len=0
7	0.050645	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=104/26624, ttl=1 (no response found!)
8	0.052924	192.168.1.254	192.168.1.64	ICMP	98 Time-to-live exceeded (Time to live exceeded in transit)

> Frame 1: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
> Ethernet II, Src: IntelCor_85:bd:d5 (e0:94:67:85:bd:d5), Dst: Actionte_fe:fa:d0 (9c:1e:95:fe:fa:d0)
v Internet Protocol Version 4, Src: 192.168.1.64, Dst: 128.119.245.12
0100 = Version: 4
.... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 56
Identification: 0x5af7 (23287)
v Flags: 0x0000
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: 255
Protocol: ICMP (1)
Header checksum: 0x2961 [validation disabled]
[Header checksum status: Unverified]
Source: 192.168.1.64
Destination: 128.119.245.12

Figure #1: First ICMP Echo Request

1. The IP address of my computer is 192.168.1.64
2. The value of the upper layer protocol is ICMP (0x01)
3. The header has 20 bytes as indicated by the Header Length field. The total packet length is 56 bytes as indicated by the Total Length field, leaving 36 bytes for the payload.
4. No. This IP datagram has not been fragmented as indicated by the More Fragments bit.

1	0.000000	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=103/26368, ttl=255 (reply in 9)
7	0.050645	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=104/26624, ttl=1 (no response found!)
10	0.101476	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=105/26880, ttl=2 (no response found!)
11	0.175072	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=106/27136, ttl=3 (no response found!)
14	0.226221	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=107/27392, ttl=4 (no response found!)
16	0.276801	192.168.1.64	128.119.245.12	ICMP	70 Echo (ping) request id=0x0001, seq=108/27648, ttl=5 (no response found!)

> Ethernet II, Src: IntelCor_85:bd:d5 (e0:94:67:85:bd:d5), Dst: Actionte_fe:fa:d0 (9c:1e:95:fe:fa:d0)
v Internet Protocol Version 4, Src: 192.168.1.64, Dst: 128.119.245.12
0100 = Version: 4
.... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 56
Identification: 0x5af8 (23288)
v Flags: 0x0000
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)
Header checksum: 0x2761 [validation disabled]
[Header checksum status: Unverified]
Source: 192.168.1.64
Destination: 128.119.245.12
> Internet Control Message Protocol

Figure #2: Second ICMP Echo Request

5. The fields that always change are:
 - a. ID

ELEC 331 Assignment 3

- b. TTL
 - c. Header checksum
- 6.
- a. The fields that stay constant are:
 - i. Version – Same version of IP (IPv4)
 - ii. Header Length – All packets are ICMP, so header length does not change
 - iii. Differentiated Services – All packets are ICMP, so same service class
 - iv. Source and Destination IP – Same host and same destination
 - v. Upper Layer Protocol – All packets are ICMP packets
 - b. The fields that must stay constant are:
 - i. Same as a).
 - c. The fields that must change are:
 - i. ID – IP packets must have different ID
 - ii. TTL – This is incremented at every router
 - iii. Header checksum – Since TTL changes, the checksum must also change
7. The pattern I observe is that the value of the ID field increments after each ICMP request.

```
555 55.894689 192.168.1.254 192.168.1.64 ICMP 590 Time-to-live exceeded (Time to live exceeded in transit)
496 53.383763 192.168.1.254 192.168.1.64 ICMP 590 Time-to-live exceeded (Time to live exceeded in transit)
358 33.070541 192.168.1.254 192.168.1.64 ICMP 590 Time-to-live exceeded (Time to live exceeded in transit)
309 30.552036 192.168.1.254 192.168.1.64 ICMP 590 Time-to-live exceeded (Time to live exceeded in transit)
181 9.698254 192.168.1.254 192.168.1.64 ICMP 98 Time-to-live exceeded (Time to live exceeded in transit)
161 7.771867 192.168.1.254 192.168.1.64 ICMP 120 Destination unreachable (Port unreachable)
138 7.189526 192.168.1.254 192.168.1.64 ICMP 98 Time-to-live exceeded (Time to live exceeded in transit)
127 6.271890 192.168.1.254 192.168.1.64 ICMP 120 Destination unreachable (Port unreachable)
87 4.703022 192.168.1.254 192.168.1.64 ICMP 120 Destination unreachable (Port unreachable)

> Ethernet II, Src: Actionte_fe:fa:d0 (9c:1e:95:fe:fa:d0), Dst: IntelCor_85:bd:d5 (e0:94:67:85:bd:d5)
> Internet Protocol Version 4, Src: 192.168.1.254, Dst: 192.168.1.64
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
  Total Length: 576
  Identification: 0x5a32 (23090)
  > Flags: 0x0000
    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: 64
  Protocol: ICMP (1)
  Header checksum: 0x993c [validation disabled]
  [Header checksum status: Unverified]
  Source: 192.168.1.254
  Destination: 192.168.1.64
> Internet Control Message Protocol
```

Figure #3: First ICMP Echo Response – TTL Expired

- 8. The value of the ID field is 23090 and the value of the TTL field is 64.
- 9. The value of the ID field changes as it gets incremented. The value of the TTL field does not change however because this is all for the “first-hop router.”

ELEC 331 Assignment 3

84	16.418067	216.140.10.30	192.168.1.102	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
85	16.438258	67.99.58.194	192.168.1.102	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
86	16.443310	192.168.1.102	128.59.23.100	ICMP	98 Echo (ping) request id=0x0300, seq=29955/885, ttl=12 (no response found!)
87	16.463382	192.168.1.102	128.59.23.100	ICMP	98 Echo (ping) request id=0x0300, seq=30211/886, ttl=13 (reply in 89)
88	16.468603	128.59.1.41	192.168.1.102	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
93	28.442185	192.168.1.102	128.59.23.100	ICMP	562 Echo (ping) request id=0x0300, seq=30467/887, ttl=1 (no response found!)
94	28.462264	10.216.228.1	192.168.1.102	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)

```
> Frame 93: 562 bytes on wire (4496 bits), 562 bytes captured (4496 bits)
> Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
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: 0x32f9 (13049)
  v Flags: 0x00b9
    0... .. = Reserved bit: Not set
    .0... .. = Don't fragment: Not set
    ..0... .. = More fragments: Not set
    ...0 0000 1011 1001 = Fragment offset: 185
  > Time to live: 1
  Protocol: ICMP (1)
  Header checksum: 0x2a7a [validation disabled]
  [Header checksum status: Unverified]
  Source: 192.168.1.102
  Destination: 128.59.23.100
  > [2 IPv4 Fragments (2008 bytes): #92(1480), #93(528)]
```

Figure #4: Fragmented ICMP Request

91	22.952738	128.119.245.12	192.168.1.102	TCP	60 22 -> 1170 [ACK] Seq=1 Ack=21 Win=35040 Len=0
92	28.441511	192.168.1.102	128.59.23.100	IPv4	1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=32f9) [Reassembled in #93]
93	28.442185	192.168.1.102	128.59.23.100	ICMP	562 Echo (ping) request id=0x0300, seq=30467/887, ttl=1 (no response found!)
94	28.462264	10.216.228.1	192.168.1.102	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
95	28.470668	192.168.1.102	128.59.23.100	IPv4	1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=32fa) [Reassembled in #96]
96	28.471338	192.168.1.102	128.59.23.100	ICMP	562 Echo (ping) request id=0x0300, seq=30723/888, ttl=2 (no response found!)
97	28.490663	192.168.1.102	128.59.23.100	IPv4	1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=32fb) [Reassembled in #98]

```
> Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
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: 1500
    Identification: 0x32f9 (13049)
  v Flags: 0x2000, More fragments
    0... .. = Reserved bit: Not set
    .0... .. = Don't fragment: Not set
    ..1... .. = More fragments: Set
    ...0 0000 0000 0000 = Fragment offset: 0
  > Time to live: 1
  Protocol: ICMP (1)
  Header checksum: 0x077b [validation disabled]
  [Header checksum status: Unverified]
  Source: 192.168.1.102
  Destination: 128.59.23.100
  Reassembled IPv4 in frame: 93
  > Data (1480 bytes)
```

Figure #5: First fragment of the Fragmented Datagram (2000 Bytes)

- I will be using the provided trace for these questions. Here, the packet was fragmented into 2 IPv4 fragments totalling up to 2000 bytes. (Fig 4)
- The header has the “More fragments” flag set to 1, meaning it has been fragmented. Looking at the fragment offset field, we see that it is 0, meaning that it is the first fragment. (Fig 5). The unfragmented IP datagram is 2000 bytes but is 2020 bytes long due to headers. The first fragment is 1500 bytes (1480 payload and 20 header).
- The “More fragments” flag was not set and we see the offset has been correctly set. Not only that, it mentions that the first fragment is #92 (Fig 5) and second fragment is #93 (Fig 4).
- Between the two fragments, the total length, flags (more fragments and fragment offset), and checksum have changed.

ELEC 331 Assignment 3

```
211 39.164169 67.99.58.194 192.168.1.102 ICMP 70 Time-to-live exceeded (Time to live exceeded in transit)
212 39.227649 128.59.1.41 192.168.1.102 ICMP 70 Time-to-live exceeded (Time to live exceeded in transit)
218 43.467629 192.168.1.102 128.59.23.100 ICMP 582 Echo (ping) request id=0x0300, seq=40451/926, ttl=1 (no response found!)

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: 568
Identification: 0x3323 (13091)
Flags: 0x0172
0... .... = Reserved bit: Not set
.0... .... = Don't fragment: Not set
..0... .... = More fragments: Not set
...0 0001 0111 0010 = Fragment offset: 370
> Time to live: 1
Protocol: ICMP (1)
Header checksum: 0x2983 [validation disabled]
[Header checksum status: Unverified]
Source: 192.168.1.102
Destination: 128.59.23.100
[3 IPv4 Fragments (3508 bytes): #216(1480), #217(1480), #218(548)]
[Frame: 216, payload: 0-1479 (1480 bytes)]
[Frame: 217, payload: 1480-2959 (1480 bytes)]
[Frame: 218, payload: 2960-3507 (548 bytes)]
[Fragment count: 3]
[Reassembled IPv4 length: 3508]
[Reassembled IPv4 data: 0800a9c303009e03373920aaaaaaaaaaaaaaaaaaaaaaaaaaaa...]

Internet Control Message Protocol
```

Figure #6: First fragment of the Fragmented Datagram (3500 Bytes)

14. 3 fragments were created.

15. The fields that change among the fragments are:

- More fragments
- Fragment offset
- Checksum
- Length (last packet only has 568 while the other have 1500)

ELEC 331 Assignment 3

Wireshark ICMP

```
C:\WINDOWS\system32>ping -n 10 www.ust.hk

Pinging www.ust.hk.w.kunlunsl.com [64.71.142.56] with 32 bytes of data:
Reply from 64.71.142.56: bytes=32 time=9ms TTL=116
Reply from 64.71.142.56: bytes=32 time=9ms TTL=116
Reply from 64.71.142.56: bytes=32 time=11ms TTL=116
Reply from 64.71.142.56: bytes=32 time=9ms TTL=116
Reply from 64.71.142.56: bytes=32 time=23ms TTL=116
Reply from 64.71.142.56: bytes=32 time=10ms TTL=116
Reply from 64.71.142.56: bytes=32 time=12ms TTL=116
Reply from 64.71.142.56: bytes=32 time=9ms TTL=116
Reply from 64.71.142.56: bytes=32 time=11ms TTL=116
Reply from 64.71.142.56: bytes=32 time=9ms TTL=116

Ping statistics for 64.71.142.56:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 9ms, Maximum = 23ms, Average = 11ms

C:\WINDOWS\system32>
```

Figure #7: Ping results

258	33.604664	192.168.1.64	64.71.142.56	ICMP	74 Echo (ping) request	id=0x0001, seq=213/54528, ttl=128 (reply in 259)
259	33.614149	64.71.142.56	192.168.1.64	ICMP	74 Echo (ping) reply	id=0x0001, seq=213/54528, ttl=116 (request in 258)
268	34.608973	192.168.1.64	64.71.142.56	ICMP	74 Echo (ping) request	id=0x0001, seq=214/54784, ttl=128 (reply in 269)

> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 60
Identification: 0x5b03 (23299)
▼ Flags: 0x0000
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: 128
Protocol: ICMP (1)
Header checksum: 0x4f56 [validation disabled]
[Header checksum status: Unverified]
Source: 192.168.1.64
Destination: 64.71.142.56
▼ Internet Control Message Protocol
Type: 8 (Echo (ping) request)
Code: 0
Checksum: 0x4c86 [correct]
[Checksum Status: Good]
Identifier (BE): 1 (0x0001)
Identifier (LE): 256 (0x0100)
Sequence number (BE): 213 (0x00d5)
Sequence number (LE): 54528 (0xd500)
[Response frame: 259]
> Data (32 bytes)

Figure #8: ICMP PING request

1. As before, the IP address of my host is 192.168.1.64 and the IP address of the destination host is 64.71.142.56.
2. An ICMP packet does not have source and destination ports because ICMP is part of the network layer with IP instead of the transport layer with TCP / UDP. As it is an integral part of the network layer, it was designed to communicate network layer information between hosts and routers and not between application layer processes.

ELEC 331 Assignment 3

3. Referring to Fig 8, we see that the Type is 8 for Echo (ping) request and Code is 0. It has a few other fields, namely checksum, identifier, sequence number, and data fields. There are two versions of both sequence number and identifier, for little and big endian.

The checksum, sequence number, and identifier fields are all 2 bytes each.

258	33.604664	192.168.1.64	64.71.142.56	ICMP	74 Echo (ping) request	id=0x0001, seq=213/54528, ttl=128 (reply in 259)
259	33.614149	64.71.142.56	192.168.1.64	ICMP	74 Echo (ping) reply	id=0x0001, seq=213/54528, ttl=116 (request in 258)
268	34.608973	192.168.1.64	64.71.142.56	ICMP	74 Echo (ping) request	id=0x0001, seq=214/54784, ttl=128 (reply in 269)

.... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 60
Identification: 0x5b03 (23299)
▼ Flags: 0x0000
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: 116
Protocol: ICMP (1)
Header checksum: 0x5b56 [validation disabled]
[Header checksum status: Unverified]
Source: 64.71.142.56
Destination: 192.168.1.64
▼ Internet Control Message Protocol
Type: 0 (Echo (ping) reply)
Code: 0
Checksum: 0x5486 [correct]
[Checksum Status: Good]
Identifier (BE): 1 (0x0001)
Identifier (LE): 256 (0x0100)
Sequence number (BE): 213 (0x00d5)
Sequence number (LE): 54528 (0xd500)
[Request frame: 258]

Figure #9: ICMP PING reply

4. For the reply, Type is 0 for Echo (ping) reply and Code is 0. It has the same fields as the request above, namely checksum, identifier, sequence number, and data. In addition, the checksum, sequence number, and identifier fields are all 2 bytes each.

```
C:\WINDOWS\system32>tracert www.inria.fr

Tracing route to ezp3.inria.fr [128.93.162.84]
over a maximum of 30 hops:

  1  1 ms    1 ms    3 ms    192.168.1.254
  2  14 ms   6 ms    7 ms    10.27.146.1
  3  10 ms   8 ms    7 ms    154.11.12.201
  4  6 ms    7 ms    5 ms    sea-b2-link.telvia.net [213.248.74.220]
  5  6 ms    5 ms    6 ms    gtt-ic-328413-sea-b2.c.telvia.net [62.115.145.71]
  6  155 ms  144 ms  149 ms  xe-2-0-0.cr0-par7.ip4.gtt.net [213.254.230.2]
  7  149 ms  180 ms  149 ms  renater-gw-ix1.gtt.net [77.67.123.206]
  8  159 ms  154 ms  156 ms  tel-1-inria-rtr-021.noc.renater.fr [193.51.177.107]
  9  158 ms  157 ms  155 ms  inria-rocquencourt-tel-4-inria-rtr-021.noc.renater.fr [193.51.184.177]
 10  174 ms  150 ms  152 ms  unit240-reth1-vfw-ext-dc1.inria.fr [192.93.122.19]
 11  152 ms  155 ms  154 ms  ezp3.inria.fr [128.93.162.84]

Trace complete.
```

Figure #10: Tracert results

ELEC 331 Assignment 3

- As before, the IP address of my host is 192.168.1.64 and the IP address of the destination host is 128.93.162.84.
- No, if ICMP sent UDP packets instead, the protocol field in the IP header should be 0x11 for UDP (17).

185	28.918768	192.168.1.64	128.93.162.84	ICMP	106 Echo (ping) request id=0x0001, seq=223/57088, ttl=1 (no response found!)
186	28.920154	192.168.1.254	192.168.1.64	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
187	28.920647	192.168.1.64	128.93.162.84	ICMP	106 Echo (ping) request id=0x0001, seq=224/57344, ttl=1 (no response found!)

Internet Protocol Version 4, Src: 192.168.1.64, Dst: 128.93.162.84

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

Total Length: 92

Identification: 0xad89 (44425)

> Flags: 0x0000

> Time to live: 1

Protocol: ICMP (1)

Header checksum: 0x277e [validation disabled]

[Header checksum status: Unverified]

Source: 192.168.1.64

Destination: 128.93.162.84

Internet Control Message Protocol

Type: 8 (Echo (ping) request)

Code: 0

Checksum: 0xf71f [correct]

[Checksum Status: Good]

Identifier (BE): 1 (0x0001)

Identifier (LE): 256 (0x0100)

Sequence number (BE): 223 (0x00df)

Sequence number (LE): 57088 (0xdf00)

> [No response seen]

> Data (64 bytes)

Figure #11: Tracert Request

- The ICMP echo packet seems to have the same fields as the ping query packets from above.

185	28.918768	192.168.1.64	128.93.162.84	ICMP	106 Echo (ping) request id=0x0001, seq=223/57088, ttl=1 (no response found!)
186	28.920154	192.168.1.254	192.168.1.64	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
187	28.920647	192.168.1.64	128.93.162.84	ICMP	106 Echo (ping) request id=0x0001, seq=224/57344, ttl=1 (no response found!)

> Frame 186: 134 bytes on wire (1072 bits), 134 bytes captured (1072 bits) on interface 0

> Ethernet II, Src: Actionte_fe:fa:d0 (9c:1e:95:fe:fa:d0), Dst: IntelCor_85:bd:d5 (e0:94:67:85:bd:d5)

> Internet Protocol Version 4, Src: 192.168.1.254, Dst: 192.168.1.64

Internet Control Message Protocol

Type: 11 (Time-to-live exceeded)

Code: 0 (Time to live exceeded in transit)

Checksum: 0xf4ff [correct]

[Checksum Status: Good]

Internet Protocol Version 4, Src: 192.168.1.64, Dst: 128.93.162.84

Internet Control Message Protocol

Type: 8 (Echo (ping) request)

Code: 0

Checksum: 0xf71f [unverified] [in ICMP error packet]

[Checksum Status: Unverified]

Identifier (BE): 1 (0x0001)

Identifier (LE): 256 (0x0100)

Sequence number (BE): 223 (0x00df)

Sequence number (LE): 57088 (0xdf00)

> Data (64 bytes)

Figure #12: Tracert Error (TTL expired)

ELEC 331 Assignment 3

8. There are a few extra fields in the ICMP error packet – it includes the IP header and the first 8 bytes of the ICMP request the error packet corresponds to.

787	80.682014	192.93.122.19	192.168.1.64	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
826	86.102057	192.168.1.64	128.93.162.84	ICMP	106 Echo (ping) request id=0x0001, seq=253/64768, ttl=11 (reply in 827)
827	86.253956	128.93.162.84	192.168.1.64	ICMP	106 Echo (ping) reply id=0x0001, seq=253/64768, ttl=51 (request in 826)
828	86.256022	192.168.1.64	128.93.162.84	ICMP	106 Echo (ping) request id=0x0001, seq=254/65024, ttl=11 (reply in 829)
829	86.411791	128.93.162.84	192.168.1.64	ICMP	106 Echo (ping) reply id=0x0001, seq=254/65024, ttl=51 (request in 828)
830	86.413924	192.168.1.64	128.93.162.84	ICMP	106 Echo (ping) request id=0x0001, seq=255/65280, ttl=11 (reply in 831)
831	86.567969	128.93.162.84	192.168.1.64	ICMP	106 Echo (ping) reply id=0x0001, seq=255/65280, ttl=51 (request in 830)

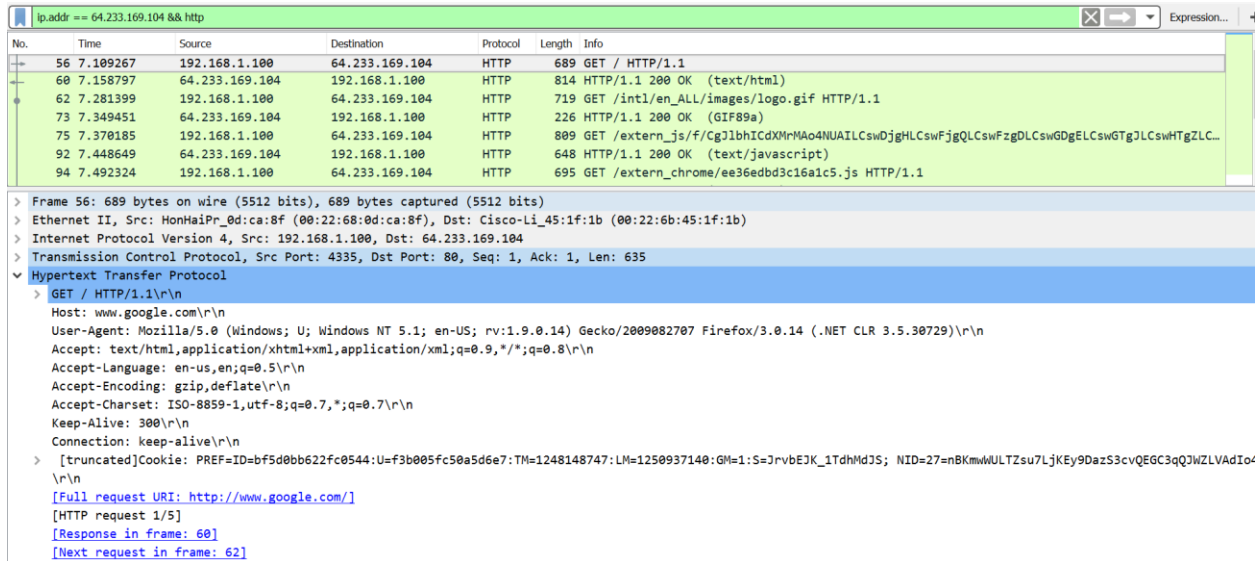
>	Frame 827: 106 bytes on wire (848 bits), 106 bytes captured (848 bits) on interface 0
>	Ethernet II, Src: Actionte_fe:fa:d0 (9c:1e:95:fe:fa:d0), Dst: IntelCor_85:bd:d5 (e8:94:67:85:bd:d5)
>	Internet Protocol Version 4, Src: 128.93.162.84, Dst: 192.168.1.64
▼	Internet Control Message Protocol
	Type: 0 (Echo (ping) reply)
	Code: 0
	Checksum: 0xff01 [correct]
	[Checksum Status: Good]
	Identifier (BE): 1 (0x0001)
	Identifier (LE): 256 (0x0100)
	Sequence number (BE): 253 (0x00fd)
	Sequence number (LE): 64768 (0xfd00)
	[Request frame: 826]
	[Response time: 151.899 ms]
>	Data (64 bytes)

Figure #13: Last three ICMP packets

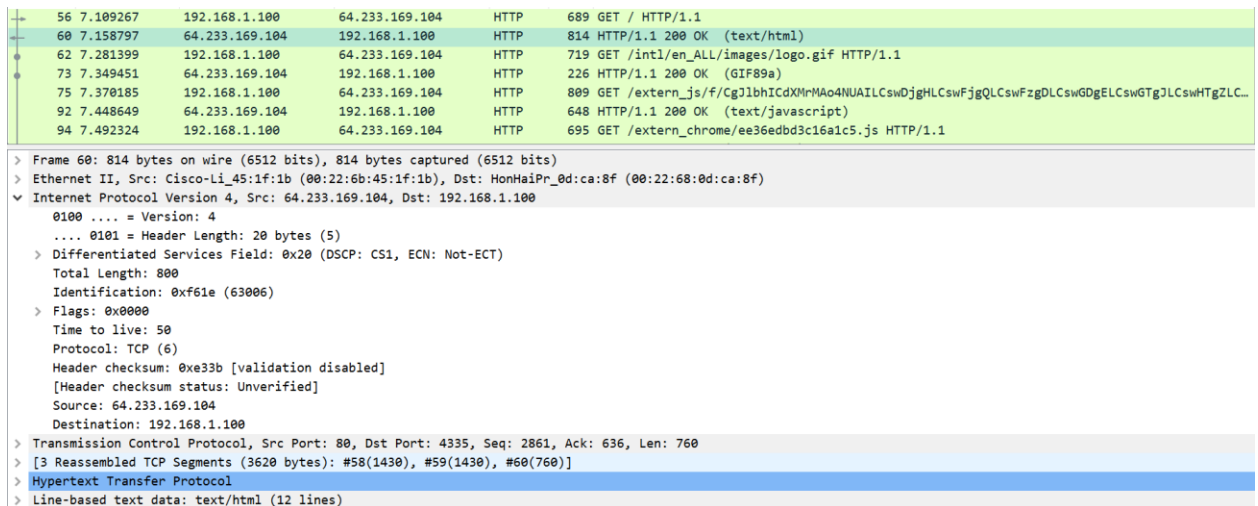
9. The last three ICMP packets are different as they have Type 0 for Echo (ping) reply instead of 11 for TTL expiration. They are different from the ICMP error packets because these datagrams have reached the destination before the TTL expired.
10. Yes. Referring to Fig 10, we see that between steps 6 and 7, there is a sudden increase in RTT. Based on the router name, I would assume that par7 is referring to Paris as the website is a French website (.fr).

ELEC 331 Assignment 3

Wireshark NAT



1. The IP address of the client is 192.168.1.100
2. Filtering above (Fig 14)
3. Source: 192.168.1.100, 4335 and Destination: 64.233.169.104, 80 (A.B.C.D, Port Number)



4. The 200 OK HTTP message was received at time t = 7.158797.
Source: 64.233.169.104, 80 and Destination: 192.168.1.100, 4335 -> reverse of 3.

ELEC 331 Assignment 3

53	7.075657	192.168.1.100	64.233.169.104	TCP	66	4335 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=4 SACK_PERM=1
54	7.108986	64.233.169.104	192.168.1.100	TCP	66	80 → 4335 [SYN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430 SACK_PERM=1 WS=64
55	7.109053	192.168.1.100	64.233.169.104	TCP	54	4335 → 80 [ACK] Seq=1 Ack=1 Win=260176 Len=0

Figure #16: TCP SYN/ACK segments

5. The TCP SYN segment was sent at t = 7.075657. The TCP ACK was received at t = 7.108986.
 - a. TCP SYN segment:
 - i. Source: 192.168.1.100, 4335 and Destination: 64.233.169.104, 80
 - b. TCP ACK response:
 - i. Source: 64.233.169.104, 80 and Destination: 192.168.1.100, 4335

85	6.069168	71.192.34.104	64.233.169.104	HTTP	689	GET / HTTP/1.1
90	6.117570	64.233.169.104	71.192.34.104	HTTP	814	HTTP/1.1 200 OK (text/html)
93	6.241357	71.192.34.104	64.233.169.104	HTTP	719	GET /intl/en_ALL/images/logo.gif HTTP/1.1
103	6.308118	64.233.169.104	71.192.34.104	HTTP	226	HTTP/1.1 200 OK (GIF89a)
106	6.330131	71.192.34.104	64.233.169.104	HTTP	809	GET /extern_js/f/CgJlbhICdXMrMAo4NUAILCswDjgHLCswFjgQLCswFzgDLCswGDgELCswGtgJLCswHTgZLC...
121	6.407366	64.233.169.104	71.192.34.104	HTTP	648	HTTP/1.1 200 OK (text/javascript)
125	6.452270	71.192.34.104	64.233.169.104	HTTP	695	GET /extern_chrome/ee36edbd3c16a1c5.js HTTP/1.1

> Frame 85: 689 bytes on wire (5512 bits), 689 bytes captured (5512 bits)
> Ethernet II, Src: Dell_4f:36:23 (00:08:74:4f:36:23), Dst: Cisco_bf:6c:01 (00:0e:d6:bf:6c:01)
> Internet Protocol Version 4, Src: 71.192.34.104, Dst: 64.233.169.104
> Transmission Control Protocol, Src Port: 4335, Dst Port: 80, Seq: 1, Ack: 1, Len: 635
▼ Hypertext Transfer Protocol
 > GET / HTTP/1.1\r\n
 Host: www.google.com\r\n User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.14) Gecko/2009082707 Firefox/3.0.14 (.NET CLR 3.5.30729)\r\n Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n Accept-Language: en-us,en;q=0.5\r\n Accept-Encoding: gzip,deflate\r\n Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n Keep-Alive: 300\r\n Connection: keep-alive\r\n > [truncated]Cookie: PREF=ID=bf5d0bb622fc0544:U=f3b005fc50a5d6e7:TM=1248148747:LM=1250937140:GM=1:S=JrvbEJK_1TdMdJS; NID=27=nBKmwULTZsu7LjKEy9DazS3cvQEGC3qQJWZLVA\r\n [Full request URI: http://www.google.com/]
 [HTTP request 1/5]
 [Response in frame: 90]
 [Next request in frame: 93]

Figure #17: NAT ISP Side of the GET request

6. The request appears on the ISP side at t = 6.069168. For this request, the destination IP and Port and source Port are the same, only the source IP has changed.
 - a. Source: 71.192.34.104, 4335 and Destination: 64.233.169.104, 80

ELEC 331 Assignment 3

No.	Time	Source	Destination	Protocol	Length	Info
56	7.109267	192.168.1.100	64.233.169.104	HTTP	689	GET / HTTP/1.1
60	7.158797	64.233.169.104	192.168.1.100	HTTP	814	HTTP/1.1 200 OK (text/html)
62	7.281399	192.168.1.100	64.233.169.104	HTTP	719	GET /intl/en_ALL/images/logo.gif HTTP/1.1
73	7.349451	64.233.169.104	192.168.1.100	HTTP	226	HTTP/1.1 200 OK (GIF89a)
75	7.370185	192.168.1.100	64.233.169.104	HTTP	809	GET /extern_js/f/Cg3lbnICdXMrMAo4NUAILCswDjgHLCswFjgQLCswFzgDLCswGDgELCswGTgJLCswHTgZLC...
92	7.448649	64.233.169.104	192.168.1.100	HTTP	648	HTTP/1.1 200 OK (text/javascript)
94	7.492324	192.168.1.100	64.233.169.104	HTTP	695	GET /extern_chrome/ee36edbd3c16a1c5.js HTTP/1.1

> Frame 56: 689 bytes on wire (5512 bits), 689 bytes captured (5512 bits)

> Ethernet II, Src: HonHaiPr_0d:ca:8f (00:22:68:0d:ca:8f), Dst: Cisco-Li_45:1f:1b (00:22:6b:45:1f:1b)

▼ Internet Protocol Version 4, Src: 192.168.1.100, Dst: 64.233.169.104

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

Total Length: 675

Identification: 0xa2ac (41644)

> Flags: 0x4000, Don't fragment

Time to live: 128

Protocol: TCP (6)

Header checksum: 0xa94a [validation disabled]

[Header checksum status: Unverified]

Source: 192.168.1.100

Destination: 64.233.169.104

Figure #18: NAT Home GET Request IP Datagram

No.	Time	Source	Destination	Protocol	Length	Info
85	6.069168	71.192.34.104	64.233.169.104	HTTP	689	GET / HTTP/1.1
90	6.117570	64.233.169.104	71.192.34.104	HTTP	814	HTTP/1.1 200 OK (text/html)
93	6.241357	71.192.34.104	64.233.169.104	HTTP	719	GET /intl/en_ALL/images/logo.gif HTTP/1.1
103	6.308118	64.233.169.104	71.192.34.104	HTTP	226	HTTP/1.1 200 OK (GIF89a)
106	6.330131	71.192.34.104	64.233.169.104	HTTP	809	GET /extern_js/f/Cg3lbnICdXMrMAo4NUAILCswDjgHLCswFjgQLCswFzgDLCswGDgELCswGTgJLCswHTgZLC...
121	6.407366	64.233.169.104	71.192.34.104	HTTP	648	HTTP/1.1 200 OK (text/javascript)
125	6.452270	71.192.34.104	64.233.169.104	HTTP	695	GET /extern_chrome/ee36edbd3c16a1c5.js HTTP/1.1

> Frame 85: 689 bytes on wire (5512 bits), 689 bytes captured (5512 bits)

> Ethernet II, Src: Dell_4f:36:23 (00:08:74:4f:36:23), Dst: Cisco_bf:6c:01 (00:0e:d6:bf:6c:01)

▼ Internet Protocol Version 4, Src: 71.192.34.104, Dst: 64.233.169.104

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

Total Length: 675

Identification: 0xa2ac (41644)

> Flags: 0x4000, Don't fragment

Time to live: 127

Protocol: TCP (6)

Header checksum: 0x022f [validation disabled]

[Header checksum status: Unverified]

Source: 71.192.34.104

Destination: 64.233.169.104

Figure #19: NAT ISP GET Request IP Datagram

- No, nothing in the HTTP GET message was changed. This can be confirmed by referring to Figures 14 and 17. Regarding the IP datagram carrying the HTTP GET, the Version, Header Length, and Flags were not changed. The only thing that was changed was checksum as both IP address and TTL changed. These changes come from a NAT table lookup.

ELEC 331 Assignment 3

87	6.099637	64.233.169.104	71.192.34.104	TCP	60 80 → 4335 [ACK] Seq=1 Ack=636 Win=7040 Len=0
88	6.117078	64.233.169.104	71.192.34.104	TCP	1484 80 → 4335 [ACK] Seq=1 Ack=636 Win=7040 Len=1430 [TCP segment of a reassembled PDU]
89	6.117407	64.233.169.104	71.192.34.104	TCP	1484 80 → 4335 [ACK] Seq=1431 Ack=636 Win=7040 Len=1430 [TCP segment of a reassembled PDU]
90	6.117570	64.233.169.104	71.192.34.104	HTTP	814 HTTP/1.1 200 OK (text/html)
91	6.118515	71.192.34.104	64.233.169.104	TCP	60 4335 → 80 [ACK] Seq=636 Ack=3621 Win=260176 Len=0
93	6.241357	71.192.34.104	64.233.169.104	HTTP	719 GET /intl/en_ALL/images/logo.gif HTTP/1.1
94	6.273849	64.233.169.104	71.192.34.104	TCP	309 80 → 4335 [PSH, ACK] Seq=3621 Ack=1301 Win=8320 Len=255 [TCP segment of a reassembled P...

```

> Frame 90: 814 bytes on wire (6512 bits), 814 bytes captured (6512 bits)
> Ethernet II, Src: Cisco_bf:6c:01 (00:0e:d6:bf:6c:01), Dst: Dell_4f:36:23 (00:08:74:4f:36:23)
> Internet Protocol Version 4, Src: 64.233.169.104, Dst: 71.192.34.104
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x20 (DSCP: CS1, ECN: Not-ECT)
    Total Length: 800
    Identification: 0xf61e (63006)
  > Flags: 0x0000
    Time to live: 51
    Protocol: TCP (6)
    Header checksum: 0x3a20 [validation disabled]
    [Header checksum status: Unverified]
    Source: 64.233.169.104
    Destination: 71.192.34.104
> Transmission Control Protocol, Src Port: 80, Dst Port: 4335, Seq: 2861, Ack: 636, Len: 760
> [3 Reassembled TCP Segments (3620 bytes): #88(1430), #89(1430), #90(760)]
> Hypertext Transfer Protocol
> Line-based text data: text/html (12 lines)

```

Figure #20: NAT ISP 200 OK IP Datagram

8. The first 200 OK HTTP message came arrived at t = 6.117570. Comparing Figures 15 and 20, it looks like the only things that are different are the Destination IP address, TTL, and checksum.
- a. Source: 64.233.169.104, 80 and Destination: 71.192.34.104, 4335

82	6.035475	71.192.34.104	64.233.169.104	TCP	66 4335 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=4 SACK_PERM=1
83	6.067775	64.233.169.104	71.192.34.104	TCP	66 80 → 4335 [SYN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430 SACK_PERM=1 WS=64
84	6.068754	71.192.34.104	64.233.169.104	TCP	60 4335 → 80 [ACK] Seq=1 Ack=1 Win=260176 Len=0

Figure #21: TCP SYN/ACK Segments

9. The TCP SYN segment was captured at t = 6.035475 and the TCP ACK segment was captured at t = 6.067775.
- a. TCP SYN:
- i. Source IP changed
- b. TCP ACK:
- i. Destination IP changed
10. NAT Table

NAT translation table	
WAN side addr	LAN side addr
71.192.34.104, 4335	192.168.1.100, 4335

ELEC 331 Assignment 3

Wireshark Ethernet and ARP

Note: I will be using the provided ethereal trace for this part of the assignment.

10	17.466468	AmbitMic_a9:3d:68	LinksysG_da:af:73	0x0800	686	IPv4
11	17.494766	LinksysG_da:af:73	AmbitMic_a9:3d:68	0x0800	60	IPv4
12	17.498935	LinksysG_da:af:73	AmbitMic_a9:3d:68	0x0800	1514	IPv4
▼ Ethernet II, Src: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)						
> Destination: LinksysG_da:af:73 (00:06:25:da:af:73)						
> Source: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)						
Type: IPv4 (0x0800)						
▼ Data (672 bytes)						
Data: 450002a000fa40008006bfc8c0a801698077f50c04220050...						
0000	00 06 25 da af 73 00 d0 59 a9 3d 68 08 00 45 00	..%..s.. Y.=h..E..				
0010	02 a0 00 fa 40 00 80 06 bf c8 c0 a8 01 69 80 77@... ..i..w...				
0020	f5 0c 04 22 00 50 65 14 99 a7 ac a5 3f b4 50 18	..."Pe...?..P..				
0030	fa f0 7e 4f 00 00 47 45 54 20 2f 65 74 68 65 72	...~0...GE T /ether				
0040	65 61 6c 2d 6c 61 62 73 2f 48 54 54 50 2d 65 74	eal-labs /HTTP-et				

Figure #22: Ethernet Frame for HTTP GET Request

1. The 48-bit Ethernet address of my computer is 00:d0:59:a9:3d:68.
2. The 48-bit destination Ethernet address is 00:06:25:da:af:73. This corresponds to the Linksys router that is used to get off the local subnet.
3. The hexadecimal value for the Frame Type is 0x0800, corresponding to the IPv4 protocol.
4. As the ASCII G is the first thing in the payload, it would appear 54 bytes from the very start of the Ethernet frame.

This is because the HTTP GET message is carried inside of a TCP segment, which is carried inside an IP datagram, which is finally carried inside the Ethernet frame. Thus, we must consider 20 bytes of header from the TCP segment, 20 bytes from the IP datagram, and finally 14 bytes for Ethernet frame's type and source and destination address. Looking at figure 22, we see that the G in GET appears at 0x37 (55), which has 54 bytes in front of it.

11	17.494766	LinksysG_da:af:73	AmbitMic_a9:3d:68	0x0800	60	IPv4
12	17.498935	LinksysG_da:af:73	AmbitMic_a9:3d:68	0x0800	1514	IPv4
13	17.500025	LinksysG_da:af:73	AmbitMic_a9:3d:68	0x0800	1514	IPv4
▼ Ethernet II, Src: LinksysG_da:af:73 (00:06:25:da:af:73), Dst: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)						
> Destination: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)						
> Source: LinksysG_da:af:73 (00:06:25:da:af:73)						
Type: IPv4 (0x0800)						
▼ Data (1500 bytes)						
Data: 456005dc8f2f4000370676f78077f50cc0a8016900500422...						
0000	00 d0 59 a9 3d 68 00 06 25 da af 73 08 00 45 00	..Y.=h.. %..s..E..				
0010	05 dc 8f 2f 40 00 37 06 76 f7 80 77 f5 0c c0 a8	.../@.7.. v..w....				
0020	01 69 00 50 04 22 ac a5 3f b4 65 14 9c 1f 50 10	..i.P..." ?..e...P..				
0030	1b 28 5e d0 00 00 48 54 54 50 2f 31 2e 31 20 32	..(^...HT TP/1.1 2				
0040	30 30 20 4f 4b 0d 0a 44 61 74 65 3a 20 53 61 74	00 [K..D ate: Sat				

ELEC 331 Assignment 3

Figure #23: Ethernet Frame for HTTP 200 OK Message

5. The Ethernet source address is 00:06:25:da:af:73, which is the Ethernet address of the Linksys router as mentioned in Question 2.
6. The Ethernet destination address is 00:d0:59:a9:3d:68, the address of my computer.
7. The hexadecimal value for the Frame Type is 0x0800, corresponding to the IPv4 protocol.
8. Similar to Question 4, as the O in OK is the first character. However, we see that it appears at 0x44 (67), which has 63 bytes in front of it. Thus, there must be additional headers or something of that sort.

```
C:\WINDOWS\system32>arp -a

Interface: 192.168.124.1 --- 0x3
  Internet Address      Physical Address      Type
  192.168.124.254       00-50-56-e3-05-c8    dynamic
  192.168.124.255       ff-ff-ff-ff-ff-ff    static
  224.0.0.2             01-00-5e-00-00-02    static
  224.0.0.22            01-00-5e-00-00-16    static
  224.0.0.251           01-00-5e-00-00-fb    static
  224.0.0.252           01-00-5e-00-00-fc    static
  239.255.3.22          01-00-5e-7f-03-16    static
  239.255.255.250       01-00-5e-7f-ff-fa    static
  255.255.255.255       ff-ff-ff-ff-ff-ff    static

Interface: 192.168.1.64 --- 0x7
  Internet Address      Physical Address      Type
  192.168.1.65          4c-8b-30-9e-cc-40    dynamic
  192.168.1.69          38-8b-59-87-c1-6a    dynamic
  192.168.1.254         9c-1e-95-fe-fa-d0    dynamic
  192.168.1.255         ff-ff-ff-ff-ff-ff    static
  224.0.0.2             01-00-5e-00-00-02    static
  224.0.0.22            01-00-5e-00-00-16    static
  224.0.0.251           01-00-5e-00-00-fb    static
  224.0.0.252           01-00-5e-00-00-fc    static
  239.255.3.22          01-00-5e-7f-03-16    static
  239.255.255.250       01-00-5e-7f-ff-fa    static
  255.255.255.255       ff-ff-ff-ff-ff-ff    static

Interface: 192.168.30.1 --- 0x14
  Internet Address      Physical Address      Type
  192.168.30.254        00-50-56-ef-48-98    dynamic
  192.168.30.255        ff-ff-ff-ff-ff-ff    static
  224.0.0.2             01-00-5e-00-00-02    static
  224.0.0.22            01-00-5e-00-00-16    static
  224.0.0.251           01-00-5e-00-00-fb    static
  224.0.0.252           01-00-5e-00-00-fc    static
  239.255.3.22          01-00-5e-7f-03-16    static
  239.255.255.250       01-00-5e-7f-ff-fa    static
  255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\WINDOWS\system32>
```

Figure #24: ARP Table

ELEC 331 Assignment 3

9.

Internet Address	IP Address
Physical Address	MAC Address
Type	Protocol Type

```

1 0.000000  AmbitMic_a9:3d:68  Broadcast  ARP  42 Who has 192.168.1.1? Tell 192.168.1.105
2 0.001018  LinksysG_da:af:73  AmbitMic_a9:3d:68  ARP  60 192.168.1.1 is at 00:06:25:da:af:73
3 0.001078  AmbitMic_a9:3d:68  LinksysG_da:af:73  0x0000  62 TPv4

> Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
v Ethernet II, Src: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  v Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Address: Broadcast (ff:ff:ff:ff:ff:ff)
      ....1. .... = LG bit: Locally administered address (this is NOT the factory default)
      ....1. .... = IG bit: Group address (multicast/broadcast)
  > Source: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)
    Type: ARP (0x0806)
  v Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0000)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)
    Sender IP address: 192.168.1.105
    Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.1.1

```

Figure #25: ARP Request Message

10.

- Source: 00:d0:59:a9:3d:68.
- Destination: ff:ff:ff:ff:ff:ff, for broadcast.

11. The hexadecimal value for the Frame Type is 0x806, corresponding to the ARP protocol.

12.

- The ARP opcode field begins 20 bytes from the very beginning of the Ethernet frame.
- Referring to fig 25, we see that the opcode is 1 (0x0001) for request.
- Yes, the ARP message contains the “Sender IP address” field.
- In the ARP message, the Target IP address field corresponds to the IP address being queried, and thus the Target MAC address field is set to 00:00:00:00:00:00 to “question” the machine whose IP is the value of the Target IP address field.

ELEC 331 Assignment 3

1	0.000000	AmbitMic_a9:3d:68	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.105
2	0.001018	LinksysG_da:af:73	AmbitMic_a9:3d:68	ARP	60	192.168.1.1 is at 00:06:25:da:af:73
3	0.001078	AmbitMic_a9:3d:68	LinksysG_da:af:73	ARP	62	IPv4

```

> Frame 2: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
  Ethernet II, Src: LinksysG_da:af:73 (00:06:25:da:af:73), Dst: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)
    Destination: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)
      Address: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)
        ....0 = LG bit: Globally unique address (factory default)
        ....0 = IG bit: Individual address (unicast)
    Source: LinksysG_da:af:73 (00:06:25:da:af:73)
      Address: LinksysG_da:af:73 (00:06:25:da:af:73)
        ....0 = LG bit: Globally unique address (factory default)
        ....0 = IG bit: Individual address (unicast)
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
  Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: LinksysG_da:af:73 (00:06:25:da:af:73)
    Sender IP address: 192.168.1.1
    Target MAC address: AmbitMic_a9:3d:68 (00:d0:59:a9:3d:68)
    Target IP address: 192.168.1.105

```

Figure #26: ARP Response Message

13.
 - a. Same answer as question 12 a), being 20 bytes from the very beginning.
 - b. Referring to fig 25, we see that the opcode is 2 (0x0002) for reply
 - c. The “answer” to the previous query appears in the “Sender MAC address” field.
14.
 - a. Source: 00:06:25:da:af:73
 - b. Destination: 00:d0:59:a9:3d:68

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	AmbitMic_a9:3d:68	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.105
2	0.001018	LinksysG_da:af:73	AmbitMic_a9:3d:68	ARP	60	192.168.1.1 is at 00:06:25:da:af:73
3	0.001028	AmbitMic_a9:3d:68	LinksysG_da:af:73	0x0800	62	IPv4
4	2.962850	AmbitMic_a9:3d:68	LinksysG_da:af:73	0x0800	62	IPv4
5	8.971488	AmbitMic_a9:3d:68	LinksysG_da:af:73	0x0800	62	IPv4
6	13.542974	CnetTech_73:8d:ce	Broadcast	ARP	60	Who has 192.168.1.117? Tell 192.168.1.104

```

> Frame 6: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
  Ethernet II, Src: CnetTech_73:8d:ce (00:80:ad:73:8d:ce), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
    Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Source: CnetTech_73:8d:ce (00:80:ad:73:8d:ce)
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
  Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: CnetTech_73:8d:ce (00:80:ad:73:8d:ce)
    Sender IP address: 192.168.1.104
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.1.117

```

Figure #26: Unreplied ARP Request Message

15. There is no reply on our trace because while the ARP request is similar to the DHCP request in that they are both broadcasts, the ARP reply is a unicast reply. As the ARP request was not

ELEC 331 Assignment 3

made by us (different source IP and MAC address), we are unable to capture the ARP reply packet.