



SAN JOSÉ STATE UNIVERSITY

Department of Computer Engineering

Network Architecture and Protocol
(CMPE208)

Internet Control Message Protocol

Group Lab 3

Group 5
Charit Upadhyay
Devika Jadhav
Pradeep Patil

Table of Contents

Introduction.....	3
Internet Control Message protocol(ICMP) Overview.....	4
ICMP Message Format.....	5
ICMP Message types	7
ICMP Message Format.....	8
ICMP Process.....	9
benefits of ICMP	10
ICMP V/s Security.....	11
ICMP Lab and Observations.....	12
Screenshots and observations Part 1.....	16
Screenshots and Observations Part 2.....	16
Contribution	24
References.....	24

Introduction

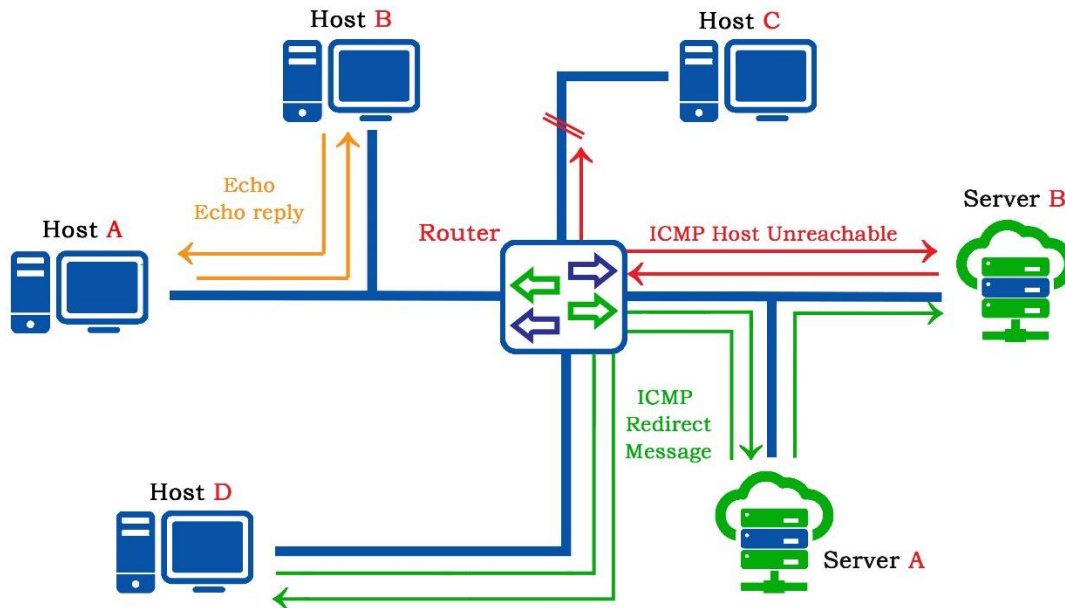


Image Reference: [created using Adobe photoshop](#)

The purpose of setting up this lab is to study the various concepts of **Internet Control Message Protocol (ICMP)**. ICMP provides a method for passing error and control messages to hosts on a TCP/IP network. ICMP is generally used by routers and hosts to send network control information to each other from a layering point of view, ICMP is a separate protocol that sits above IP and uses IP to transport messages. ICMP is an integral part of IP and all IP modules must support the ICMP protocol. With the help of following tools, we experiment and check the behavior of ICMP messages.

- **Oracle Virtual Box:** A software virtualization package that installs on an operating system as an application. VirtualBox allows additional operating systems to be installed on it, as a Guest OS, and run in a virtual environment.
- **Wireshark:** An open source network packet analyzer, which allows examining the network packet data at microscopic level.

ICMP OVERVIEW

The magnanimous scale of internet has changed the world in many ways, but its size also comes with difficult challenges. Millions of packets are being transferred and switched or the network every second. There are pretty good chances of packets getting lost or being lost due to various conditions on the network. The cause for such issues can be temporary or permanent disconnection, hardware failures, router overrun, router loops. When such things happen the host but know what went wrong. leaving the host clueless about the fate of its packet might result in nothing other than confusion leading to network instability. This problem makes way for a technique to deal with the unpredictable and problematic errors in the Network. The switching Internet Control Message Protocol (ICMP) is one such technique which is a supporting protocol that uses IP to send and receive network error messages and operational information. It is used by network devices, including routers, to send error messages and operational information.

ICMP is part of the Internet protocol suite as defined in **RFC 792** . ICMP typically uses IP header checksum to check or detect transmission error. ICMP mechanism provides a way to distinguish between events such as lost datagrams and incorrect destination addresses. Typically used by routers to inform the source of a particular datagram or packet about the unprecedented error in packet delivery.

ICMP messages are sent in several situations: for example, when a datagram cannot reach its destination, when the gateway does not have the buffering capacity to forward a datagram, and when the gateway can direct the host to send traffic on a shorter route. The Internet Protocol is not designed to be absolutely reliable. The purpose of these control messages is to provide feedback about problems in the communication environment, not to make IP reliable. There are still no guarantees that a datagram will be delivered or a control message will be returned. Some datagrams may still be undelivered without any report of their loss. The higher level protocols that use IP must implement their own reliability procedures if reliable communication is required. The ICMP messages typically report errors in the processing of datagrams. To avoid the infinite regress of messages about messages etc., no ICMP messages are sent about ICMP messages. Source (<https://tools.ietf.org/html/rfc792>)

A host can use IRDP to locate routers. The host sends router discovery packets and an IRDP-enabled router receives these. The server/client implementation of IRDP does not store full routing tables it just keeps track of which routers are sending the routing information. IRDP can also listen in on RIP updates or other routing protocols to find routers. If IRDP finds a routers via different methods then each discovery is given a priority.

ICMP Packet Format

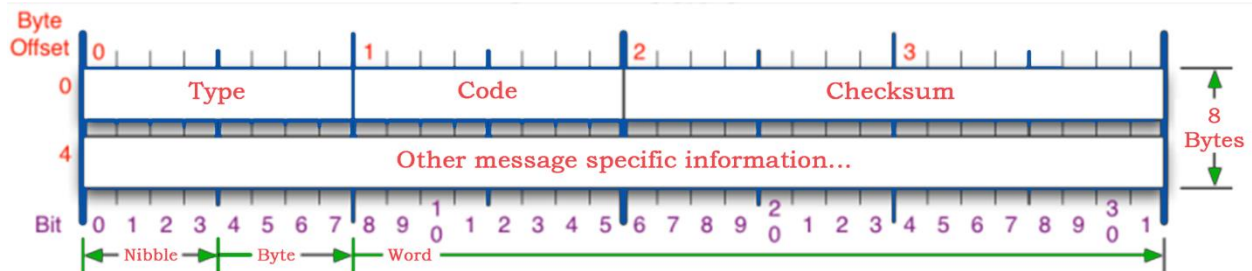


Image Reference: <https://nmap.org/book/tcpip-ref.html>

ICMP is used for error and control messages within the IP world and is very much integrated with IP. IP is not designed to be totally reliable although many common network errors are dealt with. ICMP messages give information when things do not go according to plan, however even these can get lost so for this reason no ICMP messages are sent because of previous ICMP messages going missing.

The ICMP header sits just after the IP header in the data part of the datagram. Each ICMP message has its own format and is a separate protocol. This is important to understand, particularly in firewalling. Just because you block ICMP Ping Request does not mean that you block the ICMP Ping Response, it is not part of the same connection. The ICMP message structure changes depending on Type. The general format is shown above. ICMP messages are sent using the basic IP header. The first octet of the data portion of the datagram is a ICMP type field; the value of this field determines the format of the remaining data. Any field labelled "unused" is reserved for later extensions and must be zero when sent, but receivers should not use these fields (except to include them in the checksum).

The **Type field** is used to identify the type of message and each type uses the **Code field** differently. The Variable field may contain an Identification and a Sequence number plus information such as subnet masks, IP addresses etc. again depending on the type of message.

Few well-known and most frequently encountered messages types are as shown in the table below

ICMP TYPE	ICMP CODE	DESCRIPTION
0	0	Echo Reply (used by ping)
3	0	Destination Network Unreachable
3	1	Destination Host Unreachable
3	3	Destination Port Unreachable
8	0	Echo Request (used by ping)
11	0	TTL Expired (used by trace route)

ICMP Message Types

All the ICMP messages are listed below along with any additions within the Variable field:

Type 0 - Echo Reply: This is the Echo reply from the end station which is sent as a result of the

Type 8 Echo: The Variable field is made up of a 2 octet Identifier and a 2 octet Sequence Number. The Identifier matches the Echo with the Echo Reply and the sequence number normally increments by one for each Echo sent. These two numbers are sent back to the Echo issuer in the Echo Reply.

Type 3 - Destination Unreachable: The source is told that a problem has occurred when delivering a packet. There are 5 codes, and these are as follows:

Code 0 - Net Unreachable: sent by a router to a host if the router does not know a route to a requested network.

Code 1 - Host Unreachable: sent by a router to a host if the router can see the requested network but not the destination node.

Code 2 - Protocol Unreachable: this would only occur if the destination host was reached but was not running UDP or TCP.

Code 3 - Port Unreachable: this can happen if the destination host was up and the TCP/IP was running but a service such as a web server that uses a specific port was not running.

Code 4 - Cannot Fragment: sent by a router if the router needed to fragment a packet but the Do not fragment (DF) bit was set in the IP header.

Code 5 - Source Route Failed: IP Source Routing is one of the IP Options.

Type 4 - Source Quench: the source is sending data too fast for the receiver (Code 0), the buffer has filled up, slow down!

Type 5 - Redirect: the source is told that there is another router with a better route for a packet i.e. this gateway checks its routing table and sees that another router exists on the same network with a more direct route. The Codes are assigned as follows:

Code 0 - Redirect datagrams for the network

Code 1 - Redirect datagrams for the host

Code 2 - Redirect datagrams for the Type of Service and the network

Code 3 - Redirect datagrams for the Type of Service and the host

All 4 octets of the Variable Field are used for the gateway IP address where this better router resides, and packets should therefore be sent.

Type 8 - Echo Request: this is sent by Ping (Packet Internet Groper) to a destination in order to check connectivity. The Variable field is made up of a 2 octet Identifier and a 2 octet Sequence Number. The Identifier matches the Echo with the Echo Reply and the sequence number normally increments by one for each Echo sent. These two numbers are sent back to the Echo issuer in the Echo Reply.

Type 11 - Time Exceeded: the packet has been discarded as it has taken too long to be delivered. This examines the TTL field in the IP header and the TTL exceeded code is one of the two codes used for this type. Trace under UDP, uses the TTL field to good effect. A Code value of 0 means that the Time to Live was exceeded whilst the datagram was in transit. A value of 1 means that the Fragment Reassembly Time was exceeded.

Type 12 - Parameter Problem: identifies an incorrect parameter on the datagram (Code 0). There is then a 1 octet Pointer field created in the Variable part of the ICMP packet. This pointer indicates the octet within the IP header where an error occurred. The numbering starts at 1 for the TOS field.

Type 13 - Timestamp request: this gives the round-trip time to a destination. The Variable Field is made up of two 16-bit fields and three 32-bit fields:

- **Identifier** - as with the Echo/Echo Reply
- **Sequence Number:** as with the Echo/Echo Reply
- **Originate Timestamp:** Time in milliseconds since midnight within the request as it was sent out.
- **Receive Timestamp:** Time in milliseconds since midnight as the receiver receives the message.
- **Transmit Timestamp:** Time in milliseconds since midnight within the reply as it was sent out.

The Identifier and Sequence Number field are used to match timestamp requests with replies.

Type 14 - Timestamp reply: this gives the round trip time to a particular destination.

Type 15 - Information Request: this allows a host to learn the network part of an IP address on its subnet by sending a message with the source address in the IP header filled and all zeros in the destination address field. Uses the two 16-bit Identifier and Sequence Number fields.

Type 16 - Information Reply: this is the reply containing the network portion. These two are an alternative to RARP. Uses the two 16-bit Identifier and Sequence Number fields.

Type 17 - Address mask request: request for the correct subnet mask to be used.

Type 18 - Address mask response: reply with the correct subnet mask to be used.

In summary, the ICMP header contains three fields that never change, followed by the ICMP data, then the original IP header. The first 8 bytes contain the ICMP type (major type), the second contains the code (minor code), and the third field is a checksum of the ICMP message. We need to realize that a few situations exist where ICMP will not send errors. ICMP errors will never be generated in response to an ICMP error message. If ICMP messages were sent in response to other ICMP messages, they would quickly multiply and create a storm of ICMP packets. ICMP messages will never be sent in response to a broadcast or multicast addresses either, to prevent broadcast storms.

The most useful ICMP packet is the Destination Unreachable messages, major type 3. Error messages are typically generated by routers and sent to the original source of the packet. Most of the errors are also forwarded to the application concerned with the packet that was sent. In this vein, TCP makes extensive use of ICMP

Benefits of ICMP

Diagnostic Utility

ICMP protocol helps network administrators by assisting them in diagnosing networking issues. Most issues that arise, like server outages or computer failure, are determined with two helpful commands. These commands are PING and TRACERT. An administrator uses PING to send a request from the local computer he uses to another computer or server. This request travels across the network and, once it reaches the other machine, a reply gets sent back to the original computer letting the administrator know that the communication was received. TRACERT performs the same function as PING. This tool will display the path that the request takes across the network, so the administrator can view where the breakdown on the network occurred.

Network Speed

Network speed provides users with the access on demand that they require in order to accomplish their task on the network or Internet. Many times, administrators run into situations where users complain about Internet pages, as well as network resources, taking too long to load. ICMP protocol provides administrators with the ability to send timed requests across the network, which determines if the network has a bottleneck slowing down access. Most acceptable time spans come back under 100 milliseconds; anything more usually signifies a problem either on the network or the resource a user attempts to access. This type of slow-down is called slow throughput.

Network Layer

Every network has multiple layers that make up the entire network, from the computers and servers that operate on the network, to even the pieces you do not see--like the Network layer which helps ICMP protocol actually function. The network layer builds the backbone of the Internet and all networks that transfer any type of data requests. Since the network layer plays such an important part of the network, having the ICMP protocol run on this layer allows the protocol to detect problems that arise and helps track down the source, so the administrator can correct the issue(s) right away.

Other General Advantages

- If a wrong IP address is used for configuring a client to the DNS server, an ICMP message is sent by the destination device to indicate the error.
- If a program does not allow fragmentation of its communications but it is required to communicate with a destination device, the router undertaking the fragmentation of the packet sends an ICMP message to the source device to indicate the error.
- If a client sends all communications to a particular router despite another router offering a best route, the particular router responds with the IP address of the router that provides a better route in the form of an ICMP message.
- All IP headers contain a Time to Live (TTL) value. This value is decremented as the IP packet is forwarded through each router. If a packet arrives at a router with a Time to Live (TTL) value of 1, the router cannot decrement the value any further and forward it. Instead, the router discards the packet and sends an ICMP message to indicate the expiry of the packet's TTL value.

ICMP v/s Security

Blocking ICMP Traffic for Security

Network administrators often opt to disable ICMP on network devices to evade network mapping applications used by adversaries (e.g., Nmap and Nessus scans). Unwarranted actions such as network discovery attacks, covert communication channels, and network traffic redirection could all be executed with ICMP enabled, which include but are not limited to:

Ping sweep— A type of attack that uses ICMP echo request messages to enumerate live hosts on a network.

Ping flood— Utilized to launch a denial of service attack (DoS), where the attacker sends ICMP requests in a rapid succession without waiting for the targeted system to respond. Ping floods aim to consume both incoming and outgoing bandwidth as well as utilize CPU resources to degrade the system's performance.

ICMP tunneling— A method used to establish a covert communication channel between remote systems, most times between a client and a proxy. All communications are sent via ICMP requests and replies. ICMP tunneling could be used to bypass firewall rules.

Forged ICMP redirects— Network traffic could be fraudulently redirected to an attacker via a forged ICMP redirect message. The attacker would send a ICMP redirect message, which informs a host of a direct path to a destination, to the victim that contains the IP addresses of the attacker's system. This allows an attacker to compromise network traffic via a man-in-the-middle attack or cause a DoS.

Due to all of the possible attacks involving ICMP, and the fact that TCP/IP “mostly” works even when ICMP traffic is blocked, network administrators sometimes block ICMP traffic on their firewalls as a “quick fix” security measure.

Impacts of Blocking ICMP

By disabling the ICMP protocol, diagnostics, reliability, and network performance may suffer as a result. Important mechanisms are disabled when the ICMP protocol is restricted.

Path MTU discovery (PMTUD)— . If these ICMP messages are blocked, the destination system continuously requests undelivered packets and the source system continues to resend them infinitely but to no avail, since they are too large to pass through the complete path from the source to the destination. This behavior most likely will cause a hang and is called an ICMP black hole.

Time to live (TTL). A network device with ICMP blocked will not receive type 11, time exceeded, code 0, time exceeded in transit error message — notifying the source host to increase the lifespan of the data to successfully reach the destination, if the datagram fails to reach the destination.

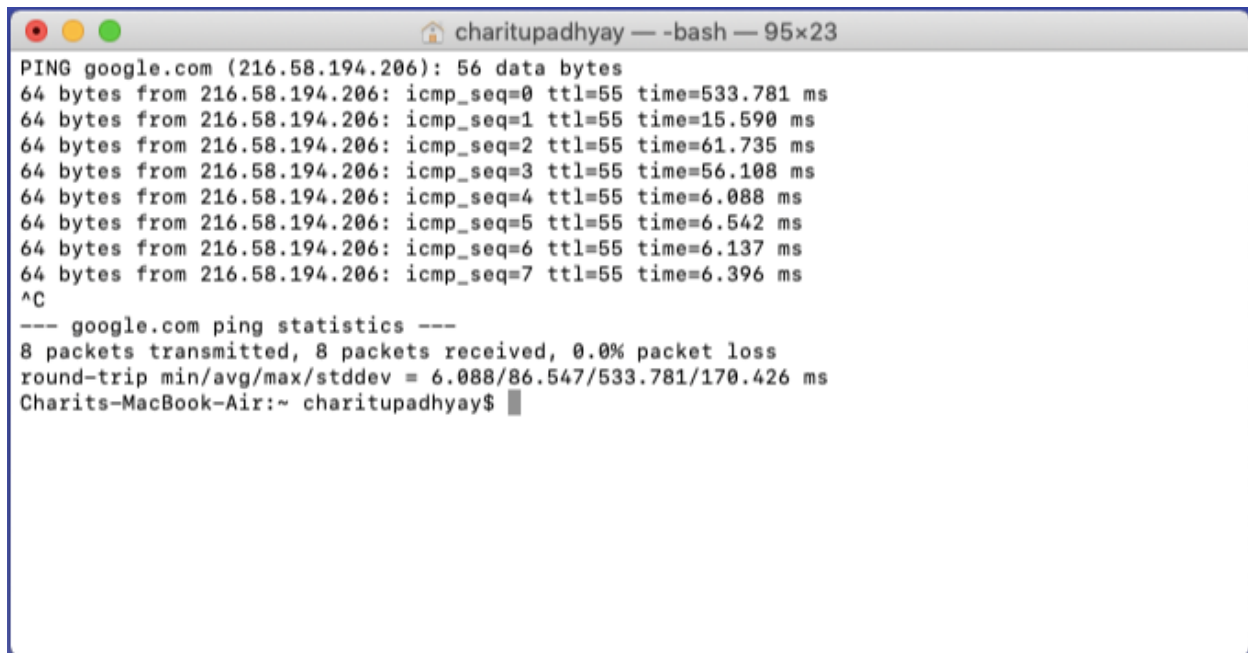
ICMP redirect— Utilized by a router to inform a host of a direct path from the host to destination. This reduces the number of hops data has to travel through to reach the destination. With ICMP disabled, the host will not be aware of the most optimal route to the destination — causing the host to send data through excessive network devices, consuming unnecessary resources which leads to the reduction of network performance.

ICMP Lab and Observations

The behavior of various ICMP messages and commands are observed using direct Linux commands on a Linux based Macbook. The packets that are sent and received are captured and studied using Wireshark. Following are the various ICMP messages and their respective packet details.

ICMP ECHO and ECHO Reply

Command : ping google.com



```
charitupadhyay — -bash — 95x23
PING google.com (216.58.194.206): 56 data bytes
64 bytes from 216.58.194.206: icmp_seq=0 ttl=55 time=533.781 ms
64 bytes from 216.58.194.206: icmp_seq=1 ttl=55 time=15.590 ms
64 bytes from 216.58.194.206: icmp_seq=2 ttl=55 time=61.735 ms
64 bytes from 216.58.194.206: icmp_seq=3 ttl=55 time=56.108 ms
64 bytes from 216.58.194.206: icmp_seq=4 ttl=55 time=6.088 ms
64 bytes from 216.58.194.206: icmp_seq=5 ttl=55 time=6.542 ms
64 bytes from 216.58.194.206: icmp_seq=6 ttl=55 time=6.137 ms
64 bytes from 216.58.194.206: icmp_seq=7 ttl=55 time=6.396 ms
^C
--- google.com ping statistics ---
8 packets transmitted, 8 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 6.088/86.547/533.781/170.426 ms
Charits-MacBook-Air:~ charitupadhyay$
```

Wireshark Packet Details

Observations:

- 32 bytes of data sent per packet
- The domain google.com resolves to IP address 216.58.194.206
- TL = 55 set by the system and is decremented by 1 each time when a packet is passed through a router.
- time = 6ms on an average, Packet's roundtrip time.
- Type: 0, as it is a ICMP reply packet
- Code: 0

Wi-Fi: en0

ICMP

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=5/1280, ttl=64 (reply in 2)
2	0.006086	216.58.194.206	10.250.52.251	ICMP	98	Echo (ping) reply id=0xe01d, seq=5/1280, ttl=55 (request in 1)
3	1.005201	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=6/1536, ttl=64 (reply in 4)
4	1.011164	216.58.194.206	10.250.52.251	ICMP	98	Echo (ping) reply id=0xe01d, seq=6/1536, ttl=55 (request in 3)
5	1.882175	fe80:1857:fa10:120...	ff02::1	ICMPv6	150	Router Advertisement from 44:2b:03:d7:f6:80
6	2.010540	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=7/1792, ttl=64 (reply in 7)
7	2.015115	216.58.194.206	10.250.52.251	ICMP	98	Echo (ping) reply id=0xe01d, seq=7/1792, ttl=55 (request in 6)
8	2.343788	10.250.52.251	216.58.194.197	TCP	54	58010 → 443 [ACK] Seq=1 Ack=1 Win=3750 Len=0
9	2.349168	216.58.194.197	10.250.52.251	TCP	66	[TCP ACKed unseen segment] 443 → 58010 [ACK] Seq=1 Ack=2 Win=10
10	3.011444	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=8/2048, ttl=64 (reply in 11)
11	3.016624	216.58.194.206	10.250.52.251	ICMP	98	Echo (ping) reply id=0xe01d, seq=8/2048, ttl=55 (request in 10)
12	4.011665	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=9/2304, ttl=64 (reply in 13)
13	4.017614	216.58.194.206	10.250.52.251	ICMP	98	Echo (ping) reply id=0xe01d, seq=9/2304, ttl=55 (request in 12)
14	4.851762	fe80:1857:fa10:120...	ff02::1	ICMPv6	150	Router Advertisement from 44:2b:03:d7:f6:80
15	5.016953	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=10/2560, ttl=64 (reply in 16)
16	5.022497	216.58.194.206	10.250.52.251	ICMP	98	Echo (ping) reply id=0xe01d, seq=10/2560, ttl=55 (request in 15)
17	6.021862	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=11/2816, ttl=64 (reply in 18)
18	6.027962	216.58.194.206	10.250.52.251	ICMP	98	Echo (ping) reply id=0xe01d, seq=11/2816, ttl=55 (request in 17)
19	7.025065	10.250.52.251	216.58.194.206	ICMP	98	Echo (ping) request id=0xe01d, seq=12/3072, ttl=64 (reply in 20)

Frame 7: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0

Ethernet II, Src: Cisco_d7:f6:80 (44:2b:03:d7:f6:80), Dst: Apple_aa:39:66 (30:35:ad:aa:39:66)

Internet Protocol Version 4, Src: 216.58.194.206, Dst: 10.250.52.251

Internet Control Message Protocol

Type: 0 (Echo (ping) reply)

Code: 0

```

0000 30 35 ad aa 39 66 44 2b 03 d7 f6 80 08 00 45 40 05..9f0+ .....E@
0010 00 54 00 00 00 00 37 01 a8 6b d8 3a c2 ce 0a fa .T...7..k.:...
0020 34 fb 00 00 42 b4 e0 1d 00 07 5b dc ec 0e 00 04 .4..B... ..[.....
0030 aa 34 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 .4..... ..[.....
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 6'()*+,-./012345
0060 36 37 67

```

Frame (frame), 98 bytes

Packets: 155 - Displayed: 155 (100.0%)

Profile: Default

Wireshark - Packet 1 - wireshark_en0_20181102173004_oW93Og

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0

Ethernet II, Src: Apple_aa:39:66 (30:35:ad:aa:39:66), Dst: Cisco_d7:f6:80 (44:2b:03:d7:f6:80)

Internet Protocol Version 4, Src: 10.250.52.251, Dst: 216.58.194.206

Internet Control Message Protocol

Type: 8 (Echo (ping) request)

Code: 0

Checksum: 0x63ea [correct]

[Checksum Status: Good]

Identifier (BE): 57373 (0xe01d)

Identifier (LE): 7648 (0x1de0)

Sequence number (BE): 5 (0x0005)

Sequence number (LE): 1280 (0x0500)

[\[Response frame: 2\]](#)

Timestamp from icmp data: Nov 2, 2018 17:30:04.295170000 PDT

[Timestamp from icmp data (relative): 0.000084000 seconds]

Data (48 bytes)

```

0000 44 2b 03 d7 f6 80 30 35 ad aa 39 66 08 00 45 00 D+...05 ..9f..E.
0010 00 54 a4 26 00 00 40 01 fb 84 0a fa 34 fb d8 3a .T.&..@. ...4...
0020 c2 ce 08 00 63 ea e0 1d 00 05 5b dc ec 0c 00 04 ....C... ..[.....
0030 81 02 08 09 0a 0b 0c 0d 0e 0f 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 6'()*+,-./012345
0060 36 37 67

```

No.: 1 - Time: 0.000000 - Source: 10.250.52.251 - Destination: 216.58.194.206 - Protocol: ICMP - Length: 98 - Info: Echo (ping) request id=0xe01d, seq=5/1280, ttl=64 (reply in 2)

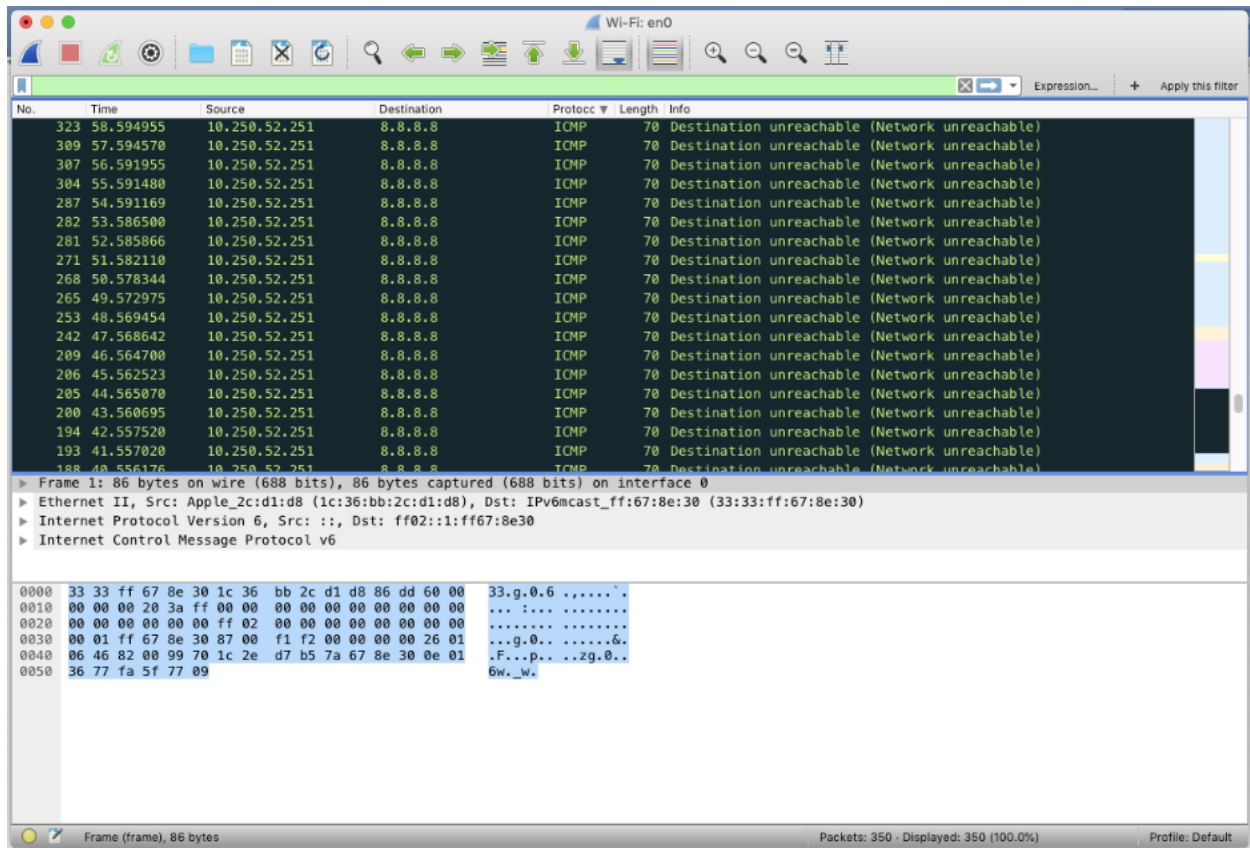
Help

Close

Code 0: Destination Unreachable

```
charitupadhyay — hping3 • sudo — 147x27
Charits-MacBook-Air:~ charitupadhyay$ sudo hping3 -a 10.250.52.251 8.8.8.8 --icmptype 3
-PING 8.8.8.8 (en0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
```

Wireshark Packet Details



Observations:

- 32 bytes of data sent per packet
- Code: 1 host unreachable.

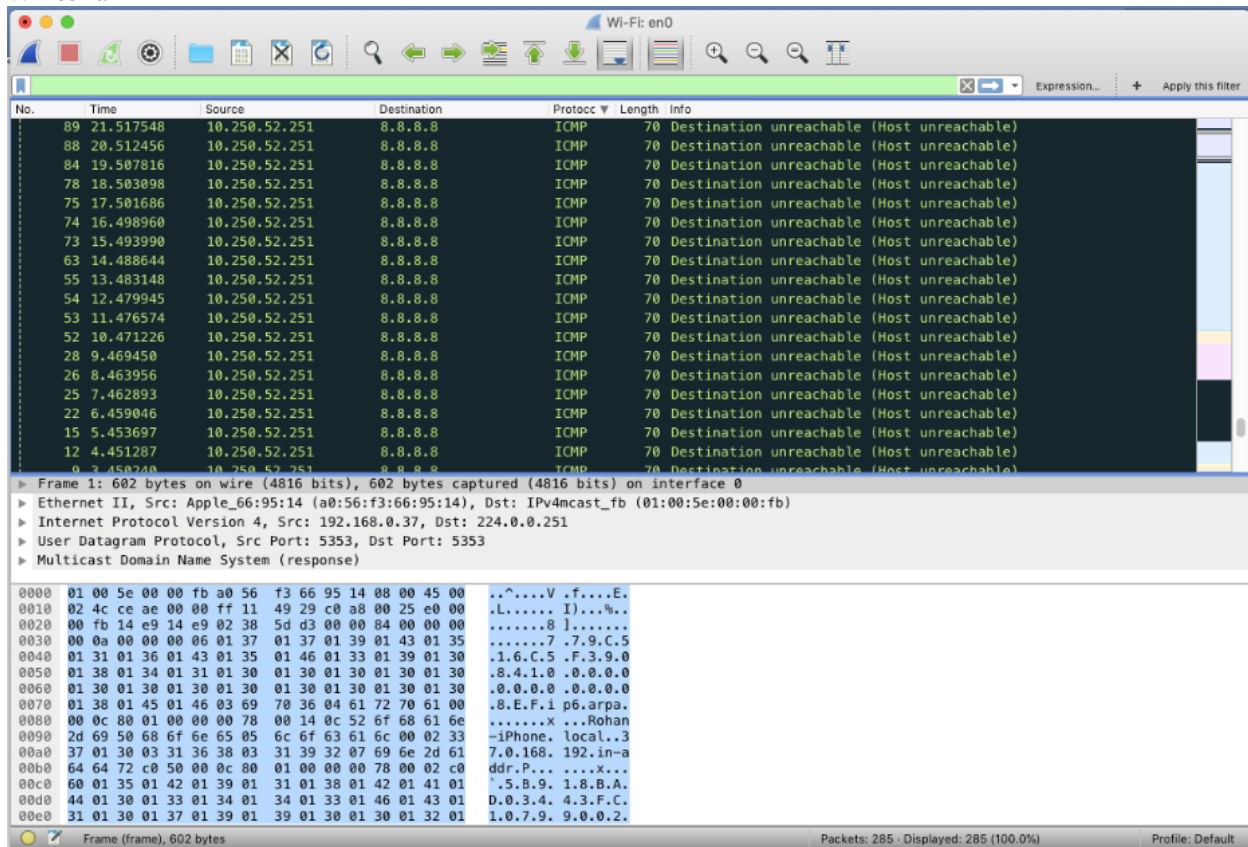
Code 2: Protocol Unreachable

It indicated that the protocol underlying transport protocol such as TCP or UDP is unavailable.

Terminal

```
Charits-MacBook-Air:~ charitupadhyay$ sudo hping3 -a 10.250.52.251 8.8.8.8 --icmpcode 2
PING 8.8.8.8 (en0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
```

Wireshark



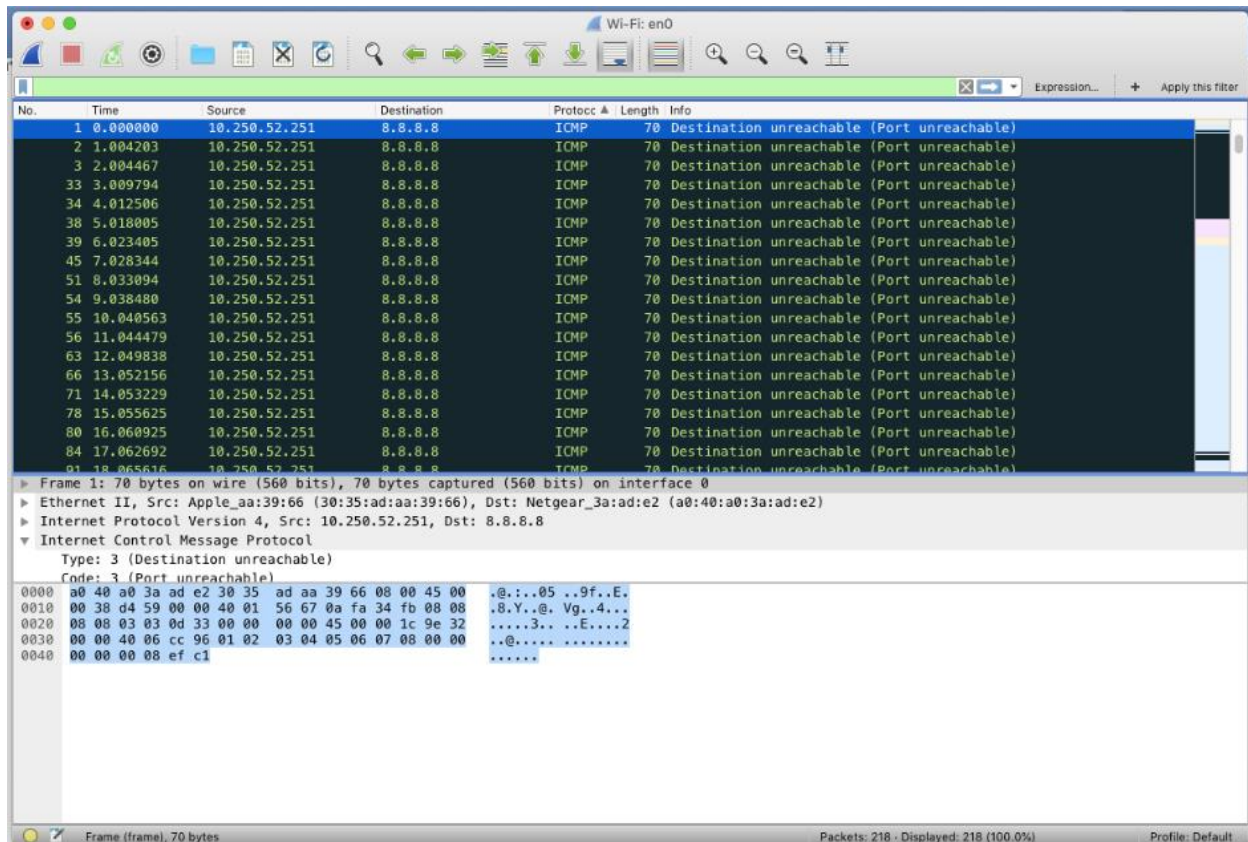
Code 3: Port Unreachable

It indicates that the application on the destination host is not active.

```
--- 8.8.8.8 hping statistic ---
64 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms

Charits-MacBook-Air:~ charitupadhyay$ sudo hping3 -a 10.250.52.251 8.8.8.8 --icmpcode 3
HPING 8.8.8.8 (en0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
```


Wireshark observations



Type 11 – Time Exceeded:

This message occurs when a TTL =1 or 0 datagram is received by the router. Router discards the datagrams with such TTL values.

Two time exceeded error codes are:

0: TTL equals 0 during transit.

1: TTL equals 1 during reassembly.

```
Charits-MacBook-Air:~ charitupadhyay$ sudo hping3 -a 10.250.52.251 8.8.8.8 --icmptype 11
HPING 8.8.8.8 (en0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
```

Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
2	1.001228	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
3	2.002464	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
8	3.003720	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
11	4.004968	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
14	5.006043	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
24	6.007159	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
27	7.007649	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
33	8.009040	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
39	9.009812	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
43	10.013849	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
46	11.018556	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
111	12.023358	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
112	13.028752	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
113	14.033091	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
114	15.036506	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
130	16.040801	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
143	17.043594	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)
148	18.046164	10.250.52.251	8.8.8.8	ICMP	70	Destination unreachable (Fragmentation needed)

Frame 1: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0

Ethernet II, Src: Apple_aa:39:66 (30:35:ad:aa:39:66), Dst: Netgear_3a:ad:e2 (a0:40:a0:3a:ad:e2)

Internet Protocol Version 4, Src: 10.250.52.251, Dst: 8.8.8.8

Internet Control Message Protocol

Type: 3 (Destination unreachable)

Code: 4 (Fragmentation needed)

0000 a0 40 a0 3a ad e2 30 35 ad aa 39 66 08 00 45 00 .@...05..9f..E.

0010 00 38 c0 c1 00 00 40 01 69 ff 0a fa 34 fb 08 08 .8....@.1...4...

0020 08 08 03 04 0d 32 00 00 00 00 45 00 00 1c 9e 3a2...E....:

0030 00 00 40 06 cc 8e 01 02 03 04 05 06 07 08 00 00 ..@.....

0040 00 00 00 08 ef c1

Type 13 and 14 – Time Stamp Request and Reply:

Timestamp request and reply messages work together. A timestamp request allows to query another for the current time. The value returned is the number of milliseconds since midnight, Coordinated Universal Time.

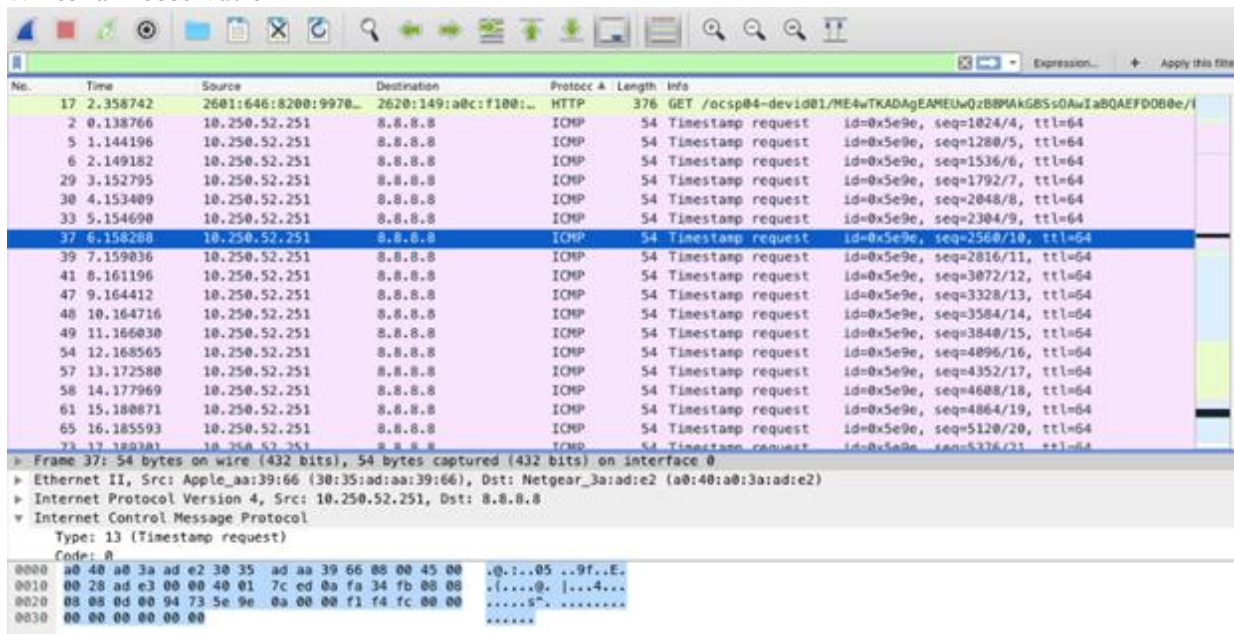
The process for time resolution:

- The requestor stamps the originate time and sends the query.
- The replying system stamps the receive time when it receives the query.
- The replying system stamps the transmit time when it sends the reply to the query.

Timestamp Request

```
[Charits-MacBook-Air:~ charitupadhyay$ sudo hping3 -a 10.250.52.251 8.8.8.8 --icmptype 13
HPING 8.8.8.8 (en0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
```

Wireshark observation



The screenshot shows a Wireshark capture on interface 0. The packet list displays 37 ICMP timestamp requests from source 10.250.52.251 to destination 8.8.8.8. The selected packet (No. 37) is expanded to show the Ethernet II, Internet Protocol Version 4, and Internet Control Message Protocol (Type: 13) headers. The raw packet data is shown in hexadecimal and ASCII at the bottom.

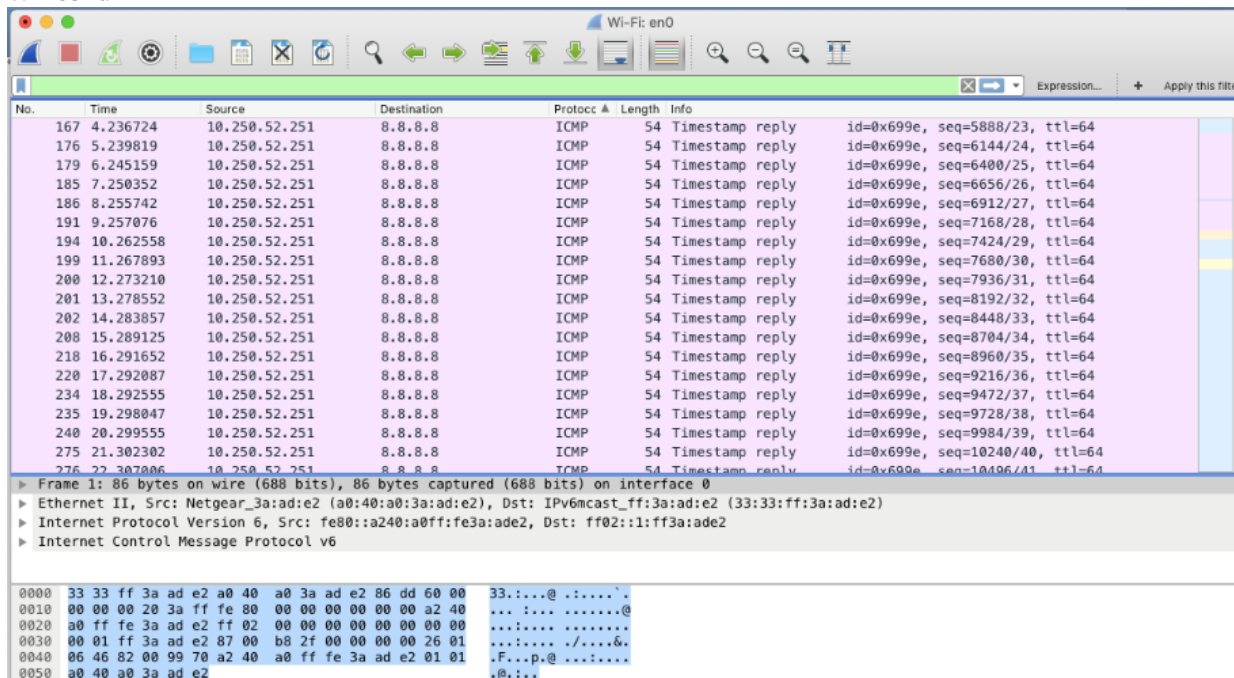
No.	Time	Source	Destination	Protocol	Length	Info
17	2.358742	2601:646:8200:9970...	2620:149:a0c:f100:...	HTTP	376	GET /ocsp04-devide01/ME4uTKADAgEAMEUwQzB8PIAkG8Ss0AwIaBQAEFD0B0e/I
2	0.138766	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=1824/4, ttl=64
5	1.144196	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=1280/5, ttl=64
6	2.149182	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=1536/6, ttl=64
29	3.152795	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=1792/7, ttl=64
30	4.153409	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=2048/8, ttl=64
33	5.154698	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=2304/9, ttl=64
37	6.158288	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=2560/10, ttl=64
39	7.159836	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=2816/11, ttl=64
41	8.161196	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=3072/12, ttl=64
47	9.164412	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=3328/13, ttl=64
48	10.164716	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=3584/14, ttl=64
49	11.166030	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=3840/15, ttl=64
54	12.168565	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=4096/16, ttl=64
57	13.172580	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=4352/17, ttl=64
58	14.177969	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=4608/18, ttl=64
61	15.180871	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=4864/19, ttl=64
65	16.185593	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=5120/20, ttl=64
73	17.189381	10.250.52.251	8.8.8.8	ICMP	54	Timestamp request id=0x5e9e, seq=5376/21, ttl=64

Frame 37: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
Ethernet II, Src: Apple_aa:39:66 (30:35:ad:aa:39:66), Dst: Netgear_3a:ad:e2 (a0:40:a0:3a:ad:e2)
Internet Protocol Version 4, Src: 10.250.52.251, Dst: 8.8.8.8
Internet Control Message Protocol
Type: 13 (Timestamp request)
Code: 0
0000 a0 40 a0 3a ad e2 30 35 ad aa 39 66 00 00 45 00 .@.1..05 ..9f..E.
0010 00 28 ad e3 00 00 40 01 7c ed 0a fa 34 fb 00 00@. |...4...
0020 08 08 0d 00 94 73 5e 9e 0a 00 00 f1 f4 fc 00 005'.....
0030 00 00 00 00 00 00

Timestamp Reply: Terminal

```
[Charits-MacBook-Air:~ charitupadhyay$ sudo hping3 -a 10.250.52.251 8.8.8.8 --icmptype 14  
HPING 8.8.8.8 (en0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
```

Wireshark



The screenshot shows a Wireshark capture on interface en0. The packet list displays 27 ICMP timestamp replies from source 8.8.8.8 to destination 10.250.52.251. The selected packet (No. 1) is expanded to show the Ethernet II, Internet Protocol Version 6, and Internet Control Message Protocol (Type: 1) headers. The raw packet data is shown in hexadecimal and ASCII at the bottom.

No.	Time	Source	Destination	Protocol	Length	Info
167	4.236724	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=5888/23, ttl=64
176	5.239819	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=6144/24, ttl=64
179	6.245159	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=6400/25, ttl=64
185	7.250352	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=6656/26, ttl=64
186	8.255742	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=6912/27, ttl=64
191	9.257076	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=7168/28, ttl=64
194	10.262558	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=7424/29, ttl=64
199	11.267893	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=7680/30, ttl=64
200	12.273210	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=7936/31, ttl=64
201	13.278552	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=8192/32, ttl=64
202	14.283857	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=8448/33, ttl=64
208	15.289125	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=8704/34, ttl=64
218	16.291652	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=8960/35, ttl=64
220	17.292087	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=9216/36, ttl=64
234	18.292555	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=9472/37, ttl=64
235	19.298047	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=9728/38, ttl=64
240	20.299555	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=9984/39, ttl=64
275	21.302302	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=10240/40, ttl=64
276	22.307086	10.250.52.251	8.8.8.8	ICMP	54	Timestamp reply id=0x699e, seq=10496/41, ttl=64

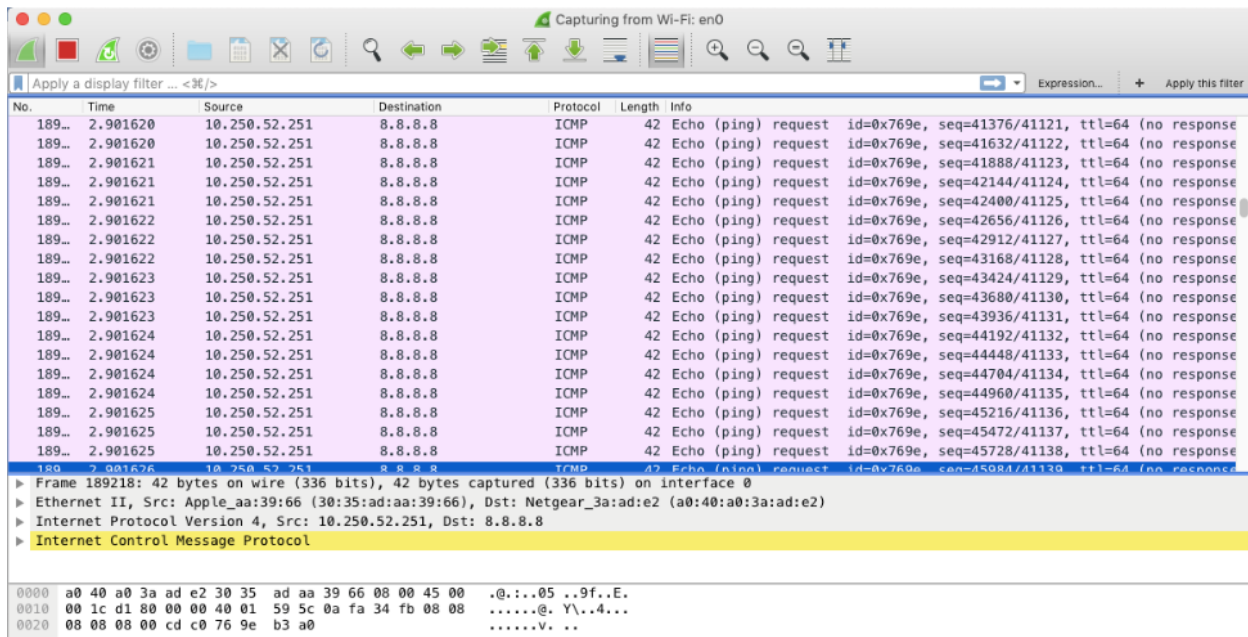
Frame 1: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface 0
Ethernet II, Src: Netgear_3a:ad:e2 (a0:40:a0:3a:ad:e2), Dst: IPv6mcast_ff:3a:ad:e2 (33:33:ff:3a:ad:e2)
Internet Protocol Version 6, Src: fe80::a240:a0ff:fe3a:ade2, Dst: ff02::1:ff3a:ade2
Internet Control Message Protocol v6
33:33:ff:3a:ad:e2 a0:40:a0:3a:ad:e2 86:dd:60:00 33:33:ff:3a:ad:e2
0000 33 33 ff 3a ad e2 a0 40 a0 3a ad e2 86 dd 60 00 33:33:ff:3a:ad:e2
0010 00 00 00 20 3a ff fe 80 00 00 00 00 00 00 a2 40@
0020 a0 ff fe 3a ad e2 ff 02 00 00 00 00 00 00 00 00
0030 00 01 ff 3a ad e2 87 00 b8 2f 00 00 00 00 26 01/.....
0040 06 46 82 00 99 70 a2 40 a0 ff fe 3a ad e2 01 01 .F...p@
0050 a0 40 a0 3a ad e2

ICMP Flooding

Terminal

```
Charits-MacBook-Air:~ charitupadhyay$ sudo hping3 -a 10.250.52.251 8.8.8.8 --icmp --flood
Password:
HPING 8.8.8.8 (en0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

Wireshark observation



No.	Time	Source	Destination	Protocol	Length	Info
189...	2.901620	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=41376/41121, ttl=64 (no response)
189...	2.901620	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=41632/41122, ttl=64 (no response)
189...	2.901621	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=41888/41123, ttl=64 (no response)
189...	2.901621	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=42144/41124, ttl=64 (no response)
189...	2.901621	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=42400/41125, ttl=64 (no response)
189...	2.901622	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=42656/41126, ttl=64 (no response)
189...	2.901622	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=42912/41127, ttl=64 (no response)
189...	2.901622	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=43168/41128, ttl=64 (no response)
189...	2.901623	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=43424/41129, ttl=64 (no response)
189...	2.901623	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=43680/41130, ttl=64 (no response)
189...	2.901623	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=43936/41131, ttl=64 (no response)
189...	2.901624	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=44192/41132, ttl=64 (no response)
189...	2.901624	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=44448/41133, ttl=64 (no response)
189...	2.901624	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=44704/41134, ttl=64 (no response)
189...	2.901624	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=44960/41135, ttl=64 (no response)
189...	2.901625	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=45216/41136, ttl=64 (no response)
189...	2.901625	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=45472/41137, ttl=64 (no response)
189...	2.901625	10.250.52.251	8.8.8.8	ICMP	42	Echo (ping) request id=0x769e, seq=45728/41138, ttl=64 (no response)

Frame 189218: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
Ethernet II, Src: Apple_aa:39:66 (30:35:ad:aa:39:66), Dst: Netgear_3a:ad:e2 (a0:40:a0:3a:ad:e2)
Internet Protocol Version 4, Src: 10.250.52.251, Dst: 8.8.8.8
Internet Control Message Protocol

0000 a0 40 a0 3a ad e2 30 35 ad aa 39 66 08 00 45 00 .@...05..9f..E.
0010 00 1c d1 80 00 00 00 01 59 5c 0a fa 34 fb 08 08@. Y\..4...
0020 08 08 08 00 cd c0 76 9e b3 a0V. ..

Conclusion

In this lab we obtained practical and detail understating of ICMP. Discussed the operation of ICMP, took a detailed look at all the ICMP Messages, Message/packet format and looked at some of the Advantages and disadvantages of using ICMP. We also covered few aspects such as security threats related to ICMP.

Contributions

Charit Upadhyay

- Command Execution and Observation
- Troubleshooting network topology
- Wireshark observation
- Equal contribution and learning on all aspects

Devika Jadhav

- ICMP message format
- ICMP Architecture
- Documentation and report formatting
- Observations on Wireshark
- Equal contribution and learning on all aspects

Pradeep Patil

- Documentation and report formatting
- ICMP Introduction
- ICMP Architecture
- Screenshots and command troubleshooting
- Equal contribution and learning on all aspects

References / Links

- <https://tools.ietf.org/html/rfc792>
- <http://searchnetworking.techtarget.com/definition/DHCP>
- [https://technet.microsoft.com/en-us/library/cc781008\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc781008(v=ws.10).aspx)
- [https://technet.microsoft.com/en-us/library/cc780760\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc780760(v=ws.10).aspx)
- Benefits <https://itstillworks.com/benefits-internet-control-message-protocol-6742787.html>
- <https://www.ipv6.com/general/icmpv6-tech-details-advantages/>
- <https://tools.ietf.org/html/rfc2131>
- <https://www.lifewire.com/what-is-dhcp-2625848>
- <https://www.thegeekstuff.com/2013/03/dhcp-basics/>
- <http://www.omniseu.com/tcpip/dhcp-dynamic-host-configuration-protocol-how-dhcp-works.php>
- <https://www.cisco.com/c/en/us/support/docs/ip/dynamic-address-allocation-resolution/27470-100.html>
- GNS: <https://www.gns3.com/>
- Wireshark: <https://www.wireshark.org/>
- ISC DHCP Server <https://help.ubuntu.com/community/isc-dhcp-server/>

- <http://www.enterprisenetworkingplanet.com/netsp/article.php/3584166/Networking-101-Understanding-and-Using-ICMP.htm>
- <http://www.rhyshaden.com/icmp.htm>
- Cisco Basic IOS commands:
https://www.cisco.com/c/en/us/td/docs/ios/12_2/configfun/command/reference/ffun_r/frf001.html