

NETWORKING PROJECT

1. Perform ping and traceroute to google.com.

Ans: **Ping** is used to test the network connectivity to a remote host and measure round-trip time.

`ping` sends ICMP Echo Request packets to the target (`google.com`), and the server replies with ICMP Echo Reply packets. It helps determine if the server is reachable and measures the time it takes for the packets to travel back and forth.

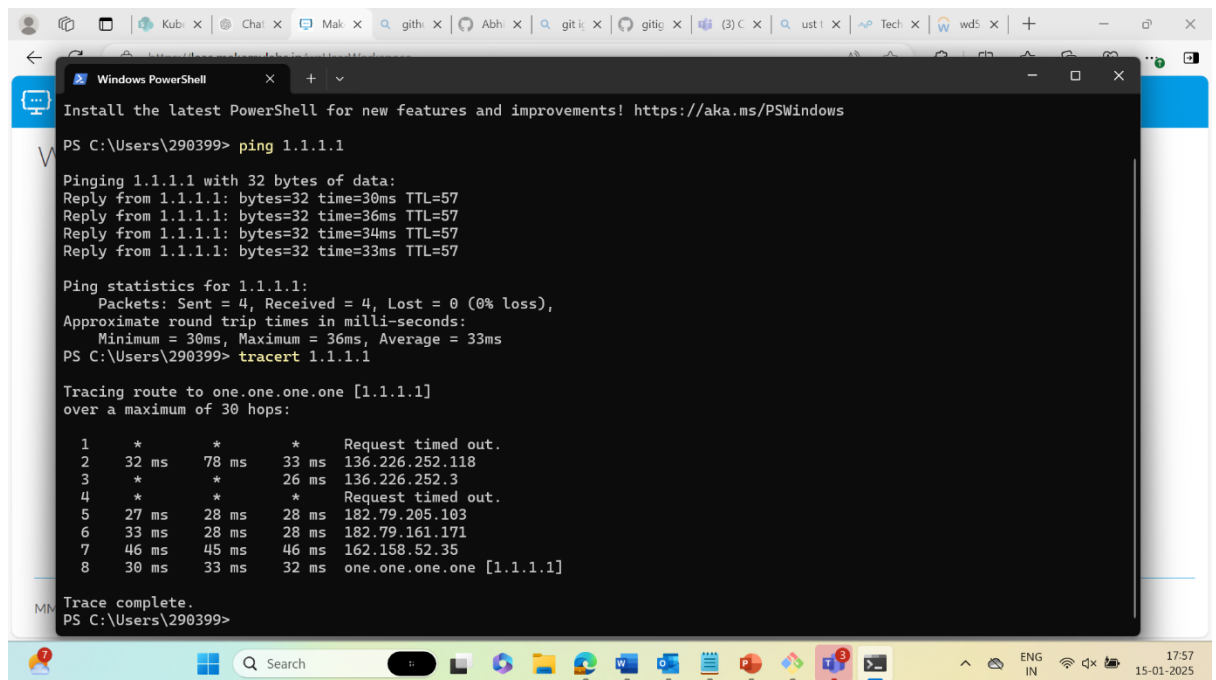
Traceroute is used to trace the path (hop-by-hop) taken by packets to reach a destination. It shows the route taken by the packets and the time it takes at each hop.

Command:

```
tracert google.com
```

Explanation:

- `tracert` sends packets with increasing time-to-live (TTL) values, and each router along the way decrements the TTL. When the TTL reaches 0, the router returns an ICMP Time Exceeded message. This helps map out each hop from your computer to the destination.



```
PS C:\Users\290399> ping 1.1.1.1

Pinging 1.1.1.1 with 32 bytes of data:
Reply from 1.1.1.1: bytes=32 time=30ms TTL=57
Reply from 1.1.1.1: bytes=32 time=36ms TTL=57
Reply from 1.1.1.1: bytes=32 time=34ms TTL=57
Reply from 1.1.1.1: bytes=32 time=33ms TTL=57

Ping statistics for 1.1.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 30ms, Maximum = 36ms, Average = 33ms
PS C:\Users\290399> tracert 1.1.1.1

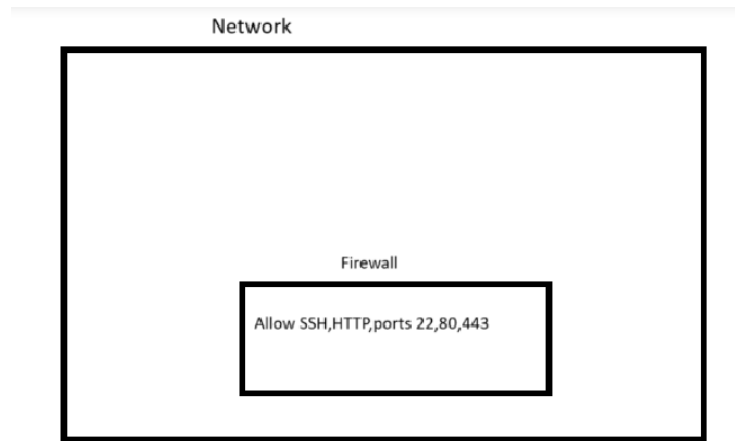
Tracing route to one.one.one.one [1.1.1.1]
over a maximum of 30 hops:

  0  *         *         *         Request timed out.
  1  32 ms     78 ms     33 ms     136.226.252.118
  2  *         *         26 ms     136.226.252.3
  3  *         *         *         Request timed out.
  4  27 ms     28 ms     28 ms     182.79.205.103
  5  33 ms     28 ms     28 ms     182.79.161.171
  6  46 ms     45 ms     46 ms     162.158.52.35
  7  30 ms     33 ms     32 ms     one.one.one.one [1.1.1.1]

Trace complete.
PS C:\Users\290399>
```

2. Design a network with firewall and open ssh,http and https port

Ans:



To design a network with a **firewall** and open **SSH (22)**, **HTTP (80)**, and **HTTPS (443)** ports, follow these steps:

Network Components:

- **Firewall:** Controls incoming and outgoing traffic.
- **Internal Network:** 192.168.2.0/24 (contains servers).
- **Web Server:** 192.168.2.2 (HTTP and HTTPS).
- **SSH Server:** 192.168.2.3 (SSH access).

To allow SSH, HTTP, and HTTPS ports using **UFW** (Uncomplicated Firewall) on Linux, follow these commands:

1. Allow SSH (port 22):

```
sudo ufw allow ssh
```

This is equivalent to:

```
sudo ufw allow 22
```

2. Allow HTTP (port 80):

```
sudo ufw allow http
```

This is equivalent to:

```
sudo ufw allow 80
```

3. Allow HTTPS (port 443):

```
sudo ufw allow https
```

This is equivalent to:

```
sudo ufw allow 443
```

To Check UFW Status:

```
sudo ufw status
```

This will show a list of allowed ports.

To Enable UFW:

If UFW is not enabled yet, you can enable it with:

```
sudo ufw enable
```

To Disable UFW (if needed):

```
sudo ufw disable
```

To Reset UFW (if you need to clear all rules):

```
sudo ufw reset
```

After running the above commands, **SSH**, **HTTP**, and **HTTPS** will be allowed, and your system will be accessible on those ports.