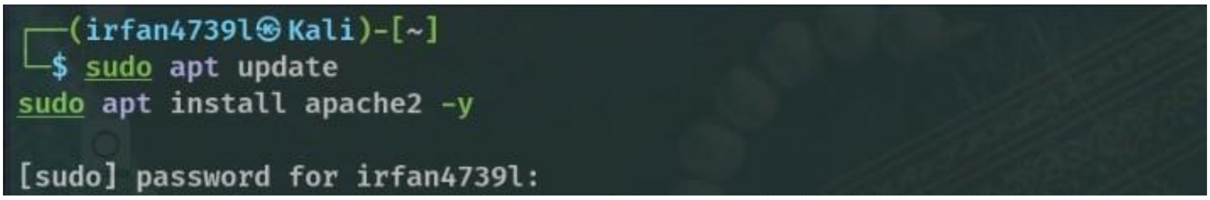
# TASK 3: Firewall & Network Security

## Setup: Install & configure apache2:

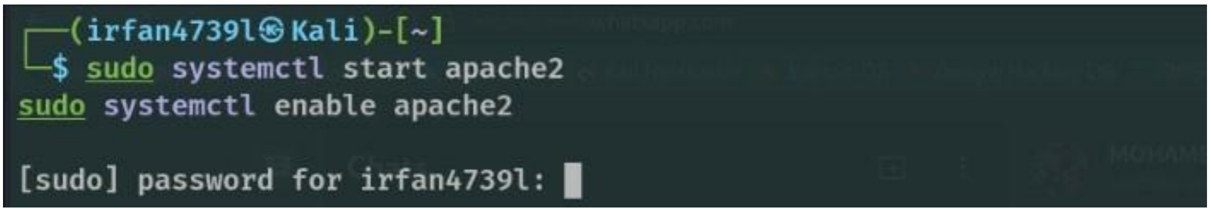
1. First, I check that my system is up-to-date and I install the apache2 using the following command: **sudo apt install apache2 -y**



```
┌──(irfan4739l㉿Kali)-[~]
└─$ sudo apt update
sudo apt install apache2 -y

[sudo] password for irfan4739l:
```

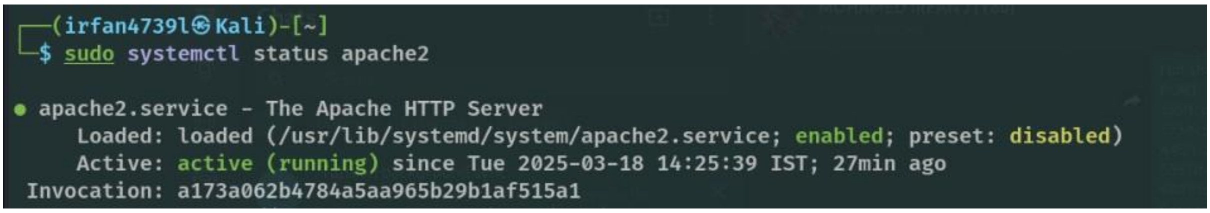
2. Next step is to start the Apache service by using command **sudo systemctl start apache2** and enable the apache2 using **sudo systemctl enable apache2**



```
┌──(irfan4739l㉿Kali)-[~]
└─$ sudo systemctl start apache2
sudo systemctl enable apache2

[sudo] password for irfan4739l: █
```

3. To check the status of Apache using the command **sudo systemctl status apache2**

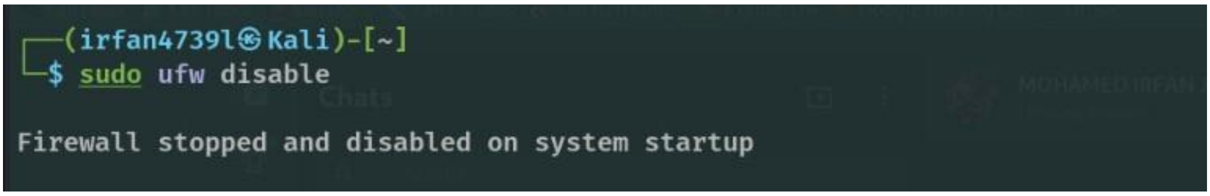


```
┌──(irfan4739l㉿Kali)-[~]
└─$ sudo systemctl status apache2

● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-03-18 14:25:39 IST; 27min ago
 Invocation: a173a062b4784a5aa965b29b1af515a1
```

## Disabling UFW to Allow All Traffic:

1. To allow all traffic, we want to disable the ufw by using the command: **sudo ufw disable**



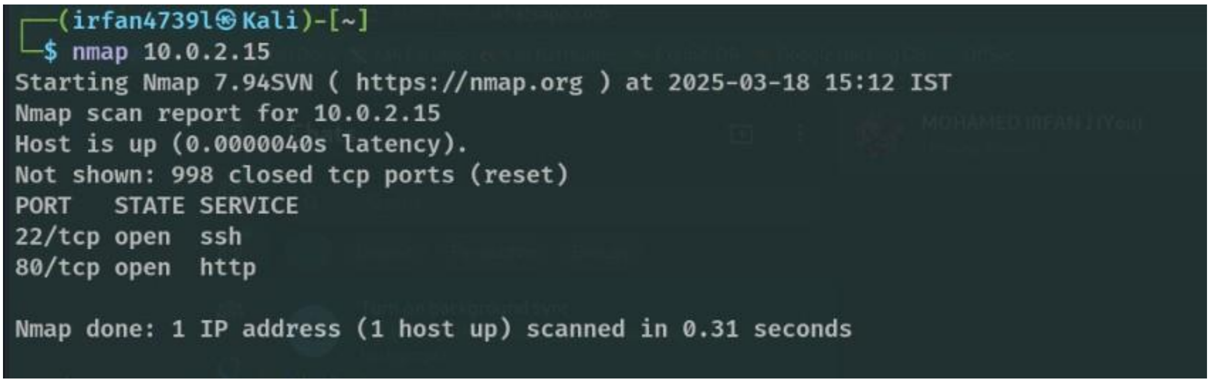
```
┌──(irfan4739l㉿Kali)-[~]
└─$ sudo ufw disable

Firewall stopped and disabled on system startup
```

## Exploitation: Use Nmap and Netcat to Scan for Open Ports & Services:

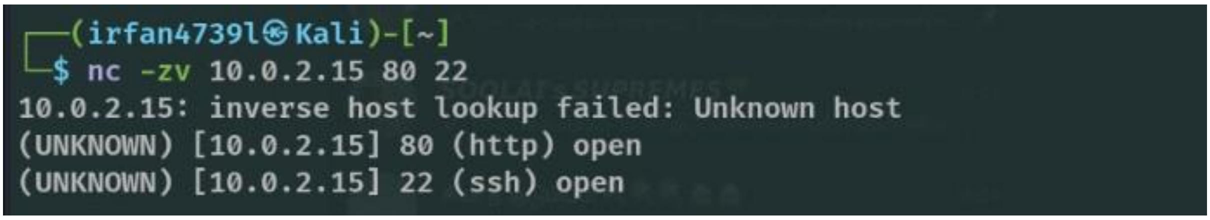
1. Now that the server is running and all traffic is allowed, we can explore how attackers might discover exposed services and open ports on the system. We will use **Nmap** and **Netcat** to scan for these open ports and services.

### Before Hardening:



```
┌──(irfan4739l㉿Kali)-[~]
└─$ nmap 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-18 15:12 IST
Nmap scan report for 10.0.2.15
Host is up (0.0000040s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
```

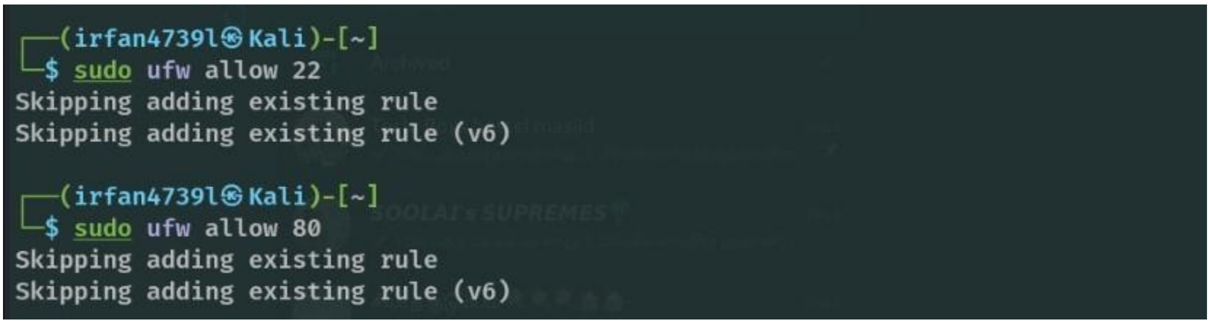


```
┌──(irfan4739l㉿Kali)-[~]
└─$ nc -zv 10.0.2.15 80 22
10.0.2.15: inverse host lookup failed: Unknown host
(UNKNOWN) [10.0.2.15] 80 (http) open
(UNKNOWN) [10.0.2.15] 22 (ssh) open
```

**Mitigation:**

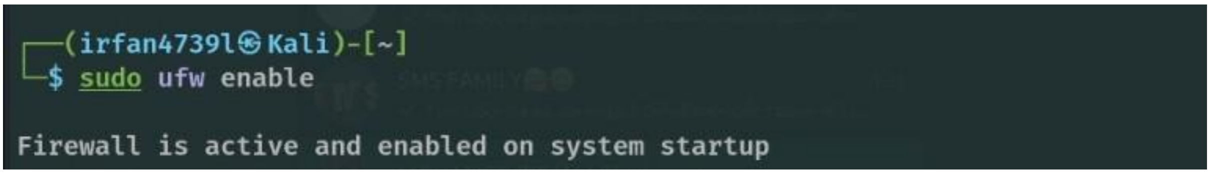
**Restrict access using ufw (only allow SSH & HTTP):**

1. Allow only SSH and HTTP traffic using the **sudo ufw allow 22   $ sudo ufw allow 80 and enable ufw**



```
┌──(irfan47391㉿Kali)-[~]
└─$ sudo ufw allow 22
Skipping adding existing rule
Skipping adding existing rule (v6)

┌──(irfan47391㉿Kali)-[~]
└─$ sudo ufw allow 80
Skipping adding existing rule
Skipping adding existing rule (v6)
```



```
┌──(irfan47391㉿Kali)-[~]
└─$ sudo ufw enable

Firewall is active and enabled on system startup
```

**Implement iptables Rules to Block Unnecessary Traffic:**

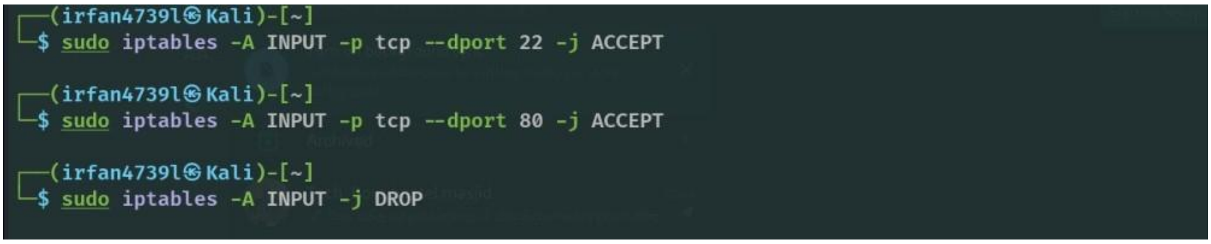
1. We Want to allow only SSH and HTTP and block all other traffic using the following commands

**sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT   # Allow SSH**

**sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT   # Allow HTTP**

**sudo iptables -A INPUT -j DROP  # Block all other incoming traffic**

and to save the iptablets using the command  **sudo iptables-save > /etc/iptables/rules.v4**

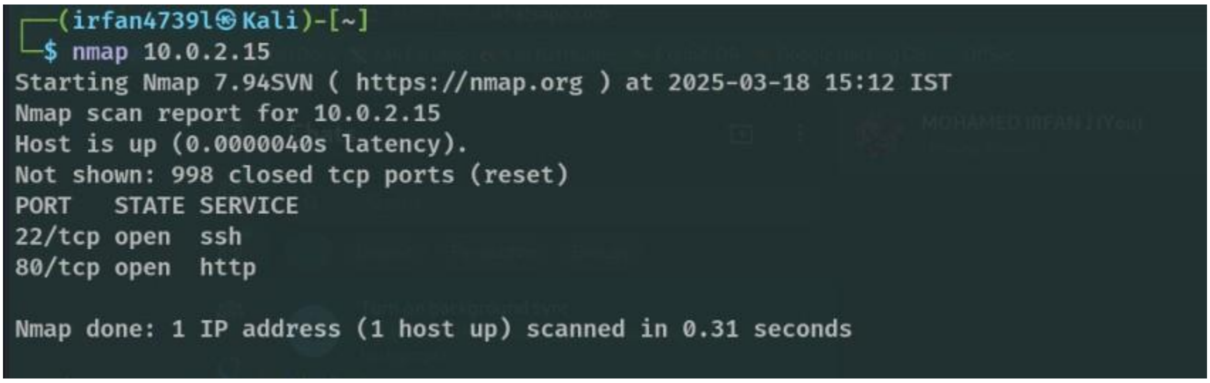


```
┌──(irfan47391㉿Kali)-[~]
└─$ sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT

┌──(irfan47391㉿Kali)-[~]
└─$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT

┌──(irfan47391㉿Kali)-[~]
└─$ sudo iptables -A INPUT -j DROP
```

## After Hardening:



```
┌──(irfan4739l㉿Kali)-[~]
└─$ nmap 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-18 15:12 IST
Nmap scan report for 10.0.2.15
Host is up (0.0000040s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
```