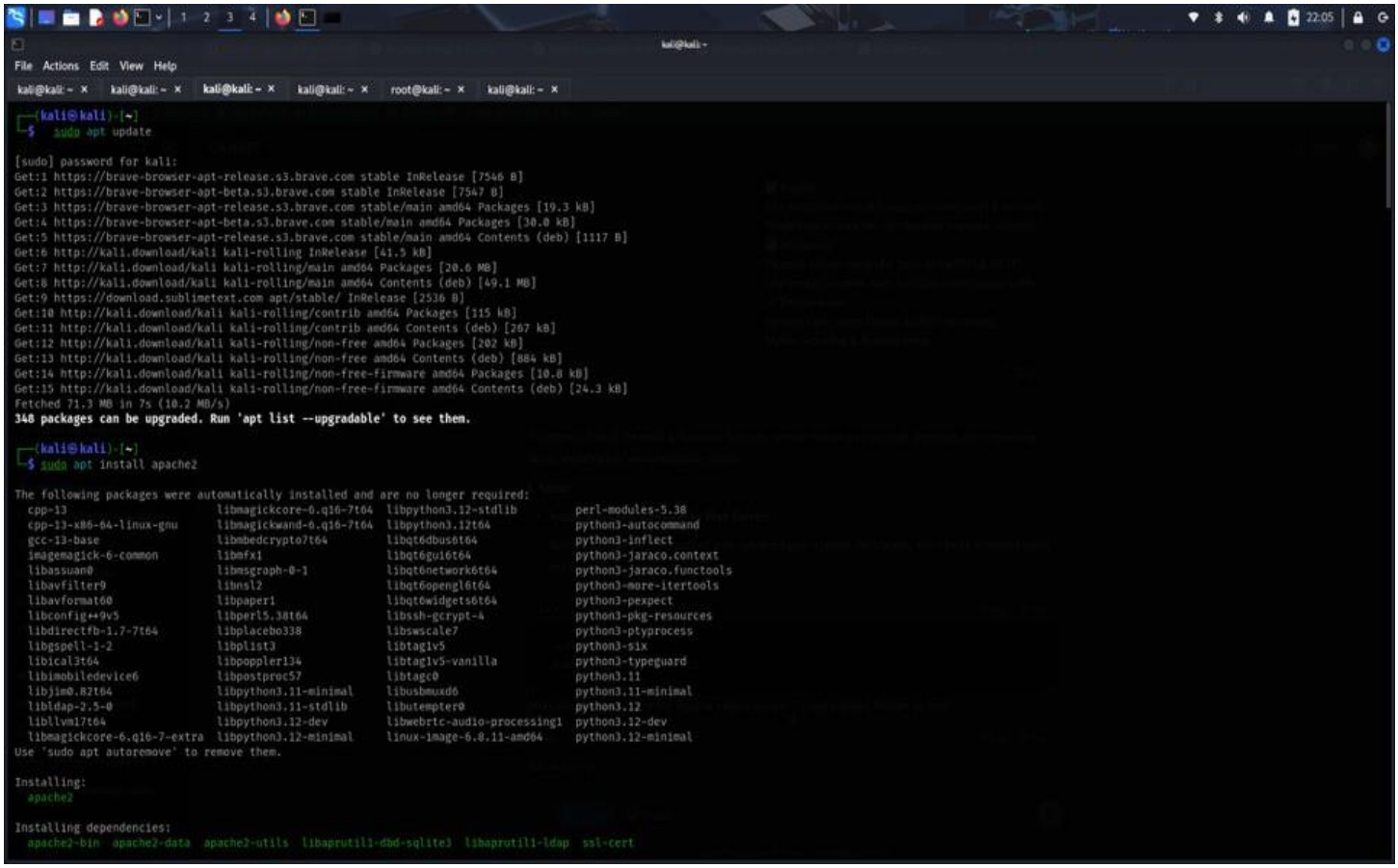
POC TASK 3

1. Setup:

- **Install and Configure Apache Web Server:**



Begin by installing the Apache2 web server on your system. On Ubuntu, this can be achieved using the following commands:

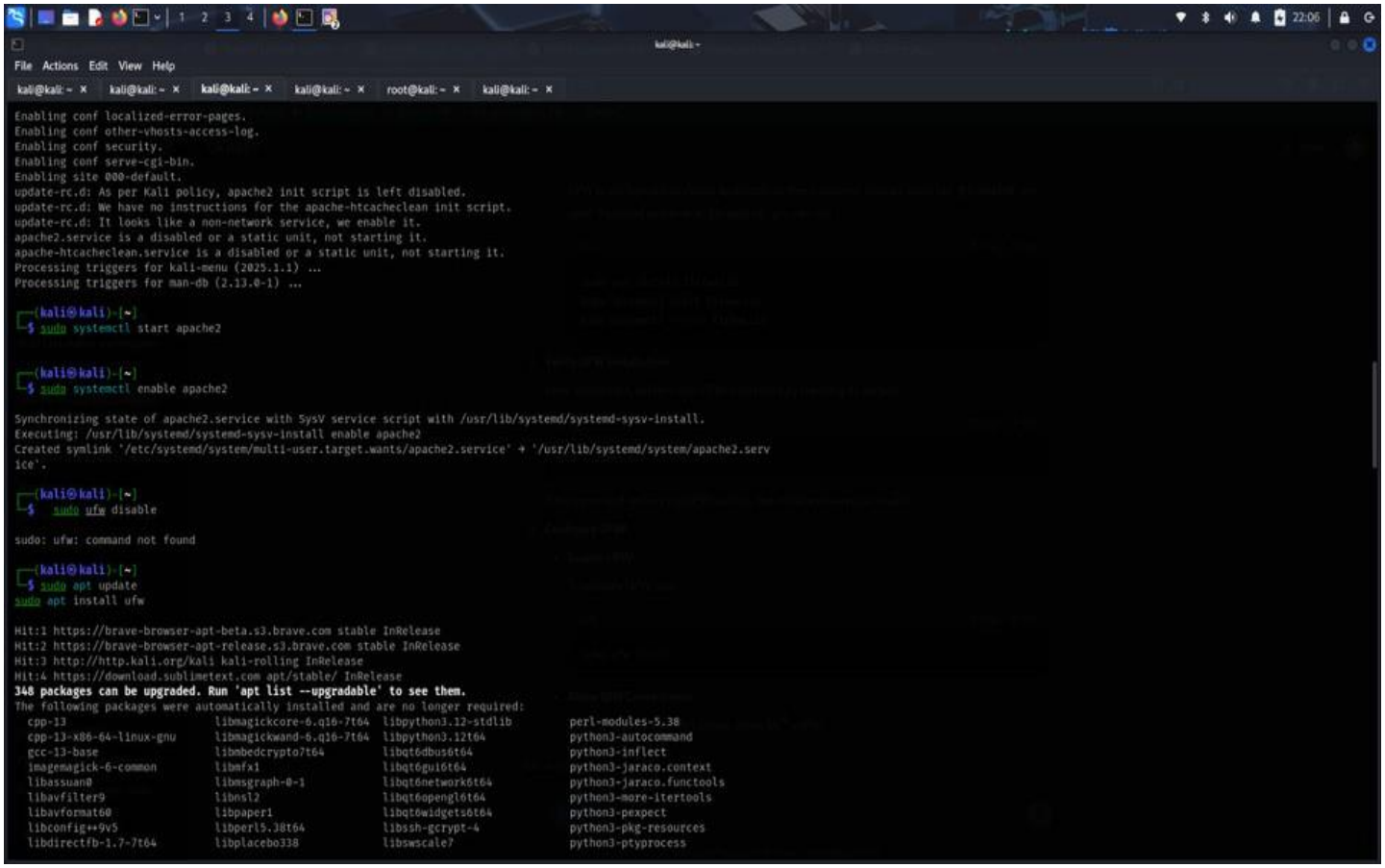
To update the package lists for available software and install Apache2, use the following commands:

Sudo apt update

The above command is essential as it refreshes the package index and ensures we fetch the latest version of Apache2:

sudo apt install apache2

- **After installation, ensure the Apache service is running and enabled to start at boot**



After the installation is complete, the next crucial step is to ensure that the Apache service is up and running. Additionally, it should be enabled to start automatically whenever the system reboots. To achieve this, use the following commands:

sudo systemctl start apache2

sudo systemctl enable apache2

- **Disable UFW to Allow All Traffic:**

To disable the Uncomplicated Firewall (UFW) and allow all incoming and outgoing traffic, I used this command

sudo ufw disable

- **Exploit:**

Scan for Open Ports and Services Using Nmap and Netcat:

With the firewall disabled, an attacker can utilize tools like Nmap and Netcat to identify open ports and running services:

- **Nmap Scan:**

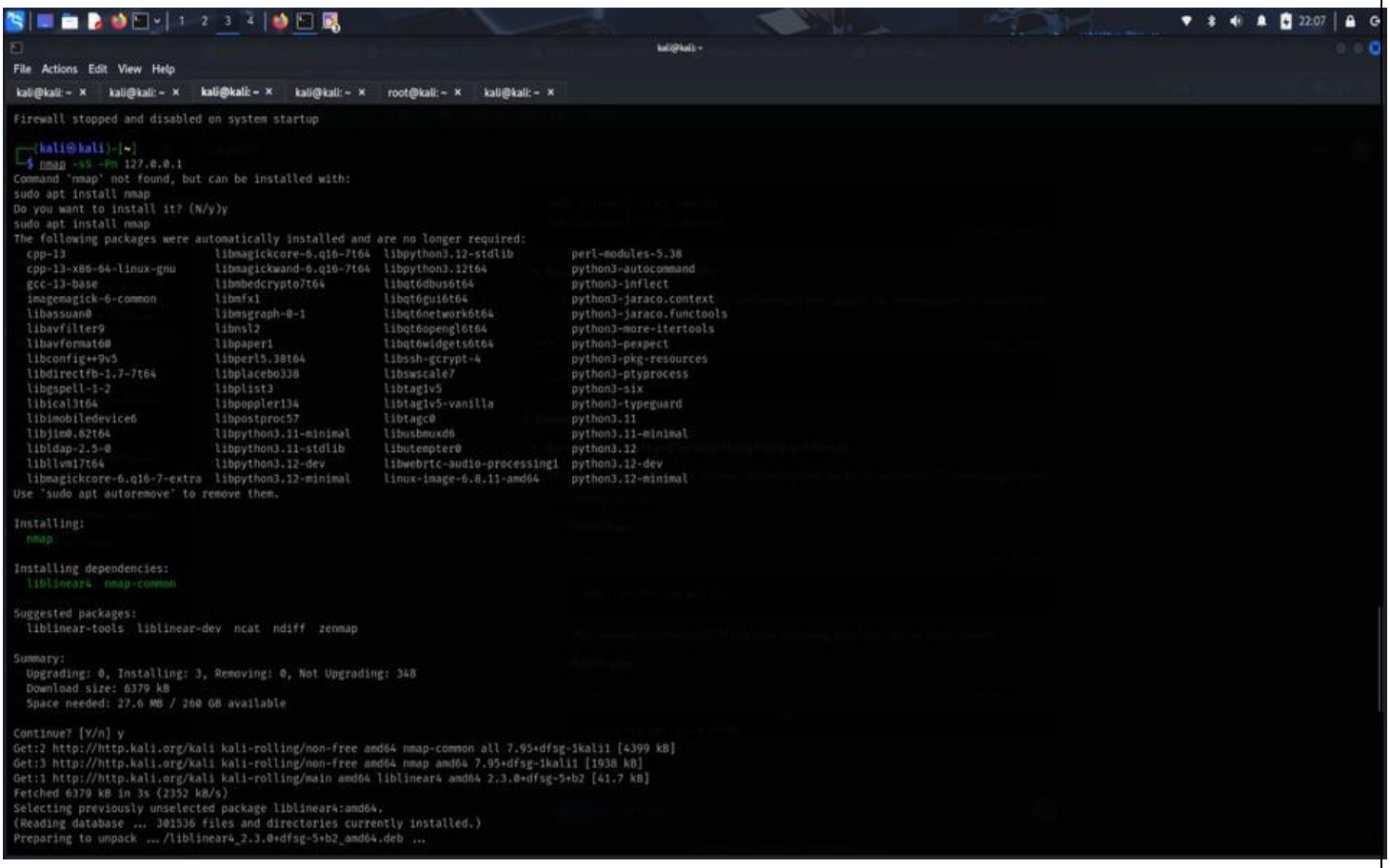
nmap -sS -Pn <target_ip>

This command performs a TCP SYN scan, detecting open ports on the target system.

- **Netcat Scan:**

To check open ports with Netcat, I used this command:

nc -zv <target_ip> 1-65535



```
Get:2 http://http.kali.org/kali kali-rolling/non-free amd64 nmap-common all 7.95+dfsg-1kali1 [4399 kB]
Get:3 http://http.kali.org/kali kali-rolling/non-free amd64 nmap amd64 7.95+dfsg-1kali1 [1938 kB]
Get:1 http://http.kali.org/kali kali-rolling/main amd64 liblinear4 amd64 2.3.0+dfsg-5+b2 [41.7 kB]
Fetched 6379 kB in 3s (2352 kB/s)
Selecting previously unselected package liblinear4:amd64.
(Reading database ... 301536 files and directories currently installed.)
Preparing to unpack .../liblinear4_2.3.0+dfsg-5+b2_amd64.deb ...
Unpacking liblinear4:amd64 (2.3.0+dfsg-5+b2) ...
Selecting previously unselected package nmap-common.
Preparing to unpack .../nmap-common_7.95+dfsg-1kali1_all.deb ...
Unpacking nmap-common (7.95+dfsg-1kali1) ...
Selecting previously unselected package nmap.
Preparing to unpack .../nmap_7.95+dfsg-1kali1_amd64.deb ...
Unpacking nmap (7.95+dfsg-1kali1) ...
Setting up liblinear4:amd64 (2.3.0+dfsg-5+b2) ...
Setting up nmap-common (7.95+dfsg-1kali1) ...
Setting up nmap (7.95+dfsg-1kali1) ...
Setcap worked! Adding configuration to environment
Processing triggers for kali-menu (2025.1.1) ...
Processing triggers for libc-bin (2.40-3) ...
Processing triggers for man-db (2.13.0-1) ...
Processing triggers for wordlists (2023.2.0) ...

┌──(kali㉿kali)-[~]
└─$ nmap -sS -Pn 127.0.0.1
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-11 20:59 IST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000030s 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 0.11 seconds

┌──(kali㉿kali)-[~]
└─$ nc -zv 127.0.0.1  1-65535

localhost [127.0.0.1] 56870 (?) open
localhost [127.0.0.1] 80 (http) open
localhost [127.0.0.1] 22 (ssh) open
```

This command checks for open TCP ports in the specified range on the target.

These scans can reveal exposed services, providing potential entry points for attackers.

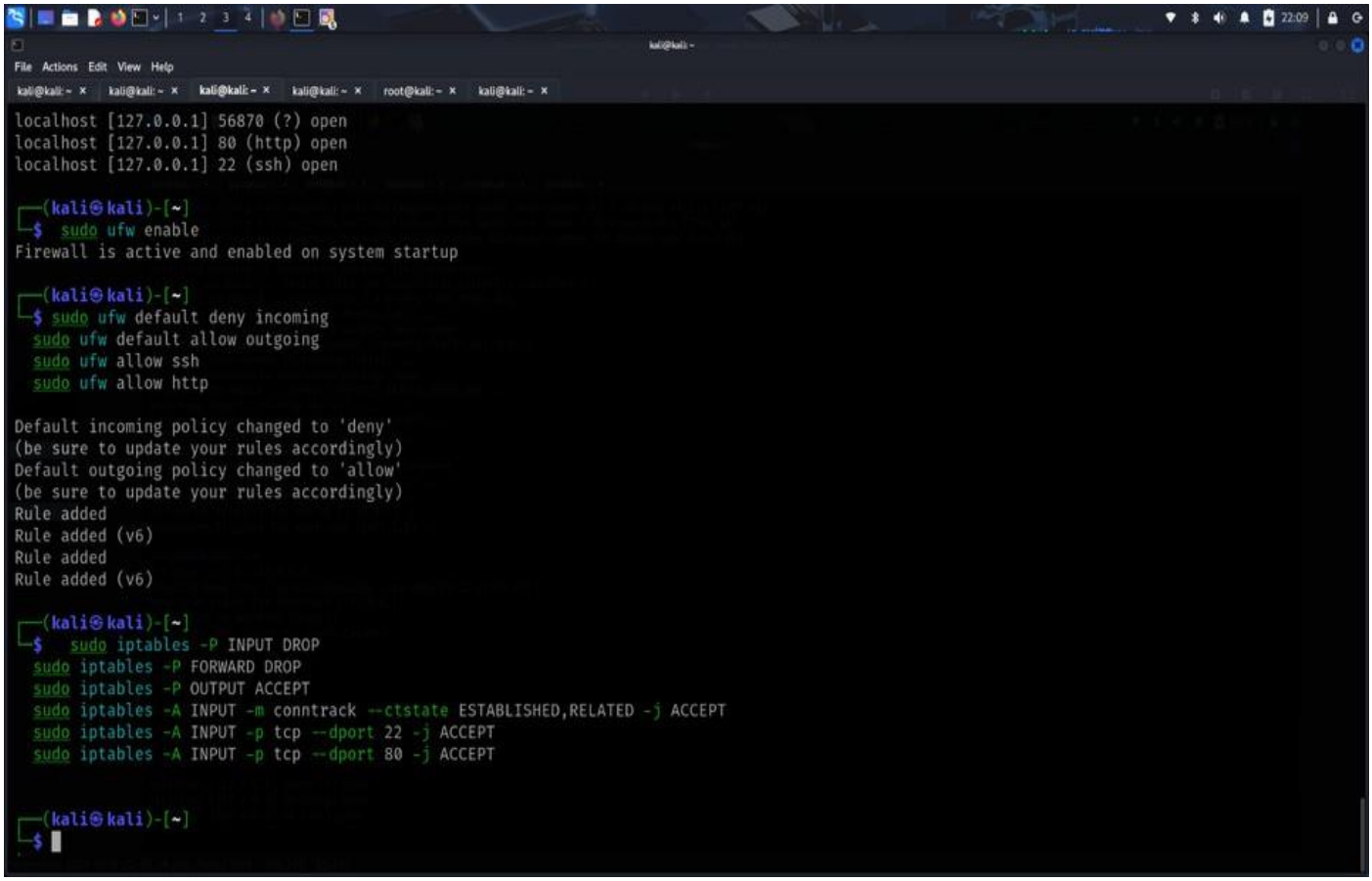
2. Mitigation:

- **Restrict Access Using UFW:**

  Re-enable UFW and configure it to allow only essential services, such as SSH (port 22) and HTTP (port 80):

```
sudo ufw enable
sudo ufw default deny incoming
sudo ufw default allow outgoing
sudo ufw allow ssh
sudo ufw allow http
```

This configuration denies all incoming tramc except for SSH and HTTP, enhancing security.



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

For more granular control, iptables can be used to define specific rules:

```
sudo iptables -P INPUT DROP
sudo iptables -P FORWARD DROP
sudo iptables -P OUTPUT ACCEPT
sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

These commands set default policies to drop incoming and forwarding tramc, accept outgoing tramc, and allow established connections along with SSH and HTTP tramc.