Report Submission

To

Task 4: Setup and Use a Firewall on Windows/Linux

Name	Yash Javiya
Submission to	Elevated Labs

Task 4 Report: Setup and Use a Firewall on Linux (UFW) Objective

To gain hands-on experience configuring a firewall on a Linux system using **UFW (Uncomplicated Firewall)**. The task involved setting and modifying rules to allow/deny traffic on specific ports and understanding how firewalls filter network traffic.

Test Environment

Component	Configuration
OS	Kali Linux (Rolling)
Target IP	127.0.0.1 / 192.168.X.X (local and bridged)
Firewall Tool	UFW (Uncomplicated Firewall)
User Privileges	Root / Sudo enabled
Ports Used for Test	22 (SSH), 23 (Telnet)
Interface Tested	eth0 (primary)
VM Platform	VMware / VirtualBox (Bridged/NAT)

Steps
Execution 1
Install and Update UFW
apt update && apt install ufw -y

```
┌──(root㉿kali)-[~]
└─# apt update && apt install ufw -y

Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [21.0 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [51.9
MB]
Fetched 72.9 MB in 30s (2,416 kB/s)
9 packages can be upgraded. Run 'apt list --upgradable' to see them.
ufw is already the newest version (0.36.2-9).
The following packages were automatically installed and are no longer require
d:
  icu-devtools           libutempter0
  libabsl20230802        libxnnpack0
  libdnnl3               python3-aioconsole
  libflac12t64           python3-dunamai
  libfuse3-3             python3-nfsclient
  libgeos3.13.0          python3-poetry-dynamic-versioning
  libglapi-mesa          python3-pywerview
  libicu-dev             python3-requests-ntlm
  libjxl0.10             python3-setproctitle
  liblbfgsb0             python3-tomlkit
  libopenh264-7          python3.12-tk
  libpoppler145          ruby-zeitwerk
  libpython3.12-minimal  sphinx-rtd-theme-common
  libpython3.12-stdlib   strongswan
  libpython3.12t64
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 9
```

Task 4th: Setup and Use a Firewall on Linux (UFW)

1. Check UFW Status

sudo ufw status

Output: Initially disabled (default on most Kali builds)

```
┌──(root㉿kali)-[~]
└─# ufw status
Status: inactive
```

2. Enable UFW

sudo ufw enable

```
┌──(root㉿kali)-[~]
└─# sudo ufw enable
Firewall is active and enabled on system startup
```

Ensures the firewall starts protecting the system immediately.

3. View Existing Rules

sudo ufw status numbered

```
┌──(root㉿kali)-[~]
└─# sudo ufw status numbered
Status: active
```

Shows a numbered list of all current UFW rules (empty initially).

4. Block Port 23 (Telnet)

sudo ufw deny 23

```
┌──(root㉿kali)-[~]
└─# sudo ufw deny 23
Rule added
Rule added (v6)
```

This simulates blocking a dangerous/unsecured service (Telnet is clear-text and insecure).

5. Allow SSH Access (Port 22)

sudo ufw allow 22/tcp

```
┌──(root㉿kali)-[~]
└─# sudo ufw allow 22/tcp
Rule added
Rule added (v6)
```

Ensures that remote access via SSH remains available.

6. Verify Rules Applied

sudo ufw status verbose

```
┌──(root㉿kali)-[~]
└─# sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
23                         DENY IN     Anywhere
22/tcp                     ALLOW IN    Anywhere
23 (v6)                    DENY IN     Anywhere (v6)
22/tcp (v6)                ALLOW IN    Anywhere (v6)
```

7. Test the Blocked Port

Used nmap locally:

nmap -p 23 localhost

```
┌──(root㉿kali)-[~]
└─# nmap -p 23 localhost
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-01 11:27 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0045s latency).
Other addresses for localhost (not scanned): ::1

PORT   STATE  SERVICE
23/tcp closed telnet

Nmap done: 1 IP address (1 host up) scanned in 0.17 seconds
```

Output: **Connection refused or timed out** (as expected)

8. Remove Test Rule (Port 23 Deny)

sudo ufw delete deny 23

```
┌──(root㉿kali)-[~]
└─# sudo ufw delete deny 23
Rule deleted
Rule deleted (v6)
```

Restores system to original open state for port 23 (if needed for reuse in other labs).

---

9. Disable UFW (if cleanup needed)

sudo ufw disable

```
┌──(root㉿kali)-[~]
└─# sudo ufw disable
Firewall stopped and disabled on system startup
```

Understanding Firewall Traffic Filtering

Concept	Description
Inbound Rules	Control incoming traffic (e.g., block Telnet on port 23)
Outbound Rules	Control outgoing traffic (e.g., prevent data exfiltration to external IPs)
Default Policy	UFW uses default DENY/ALLOW policies that can be modified
Ports	Firewalls operate by controlling TCP/UDP ports access