Task 4

In this task we will setup and understand how to use a firewall on Linux(kali), but first we need to install it our system here's the full process:

1. Run this command on your terminal
   " sudo apt update
   sudo apt install ufw -y
   sudo ufw enable "
2. This will update your linux first then install **UFW** which is Uncomplicated Firewall a tool which helps us configure and manage firewall service on linux OS.
3. The completed process should look something like fig1.

```
┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo apt update
sudo apt install ufw -y
sudo ufw enable
Hit:1 https://packages.microsoft.com/debian/12/prod bookworm InRelease
Get:2 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:3 http://kali.download/kali kali-rolling/main arm64 Packages [20.8 MB]
Get:4 http://kali.download/kali kali-rolling/main arm64 Contents (deb) [50.7 MB]
Get:5 http://kali.download/kali kali-rolling/non-free arm64 Packages [150 kB]
Get:6 http://kali.download/kali kali-rolling/non-free arm64 Contents (deb) [860 kB]
Fetched 72.6 MB in 50s (1455 kB/s)
870 packages can be upgraded. Run 'apt list --upgradable' to see them.
ufw is already the newest version (0.36.2-9).
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 870
Firewall is active and enabled on system startup
```

fig1

4. Now that we have UFW installed let's see whay preconfigured firewall rules does our system have. Fig2

```
┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
```

Now that our UFW is installed let's try and do some configuration changes to enhance security of our device:

1. First let's see which ports are listening on our device through the help of a simple command: " sudo netstat -tuln "
   Let's break down what this command does:
   - first netstat just shows listening sockets
   - -t shows tcp ports
   - -u shows udf ports
   - -l showing listening
   - -n shows port numbers instead of their name like (23 instead of telnet)

2. The output of my firewall is in fig3:

```
┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:1883          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:9392          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:5432          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:80            0.0.0.0:*               LISTEN
tcp6       0      0 ::1:1883                :::*                    LISTEN
tcp6       0      0 ::1:5432                :::*                    LISTEN
udp        0      0 0.0.0.0:42989           0.0.0.0:*
udp        0      0 192.168.31.101:3702     0.0.0.0:*
udp        0      0 239.255.255.250:3702    0.0.0.0:*
udp6       0      0 :::40414                :::*
udp6       0      0 fe80::7ded:a51c:7c:3702 :::*
udp6       0      0 ff02::c:3702            :::*
udp6       0      0 fe80::7ded:a51c:7c4:546 :::*
```

Fig3

3. This tells a lot that our preconfigured firewall is actually doing pretty ok as only the important services are listening.
4. Next are the security measures that should be taken everytime when configuring a firewall which will only help enhace the security of your device, No. 1 is to disable telnet which is not currently listnenig on my device but is still a good security measure because it can still be accesible when certain application ask for it. Simple command to disable telnet is in fig4:

```
┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo ufw deny 23
[sudo] password for kali:
Rule added
Rule added (v6)

┌──(kali㉿kali)-[~/ELI/day4]
└─$ _
```

Fig4

5. Next is to enable ssh because it is really secure and helps to remotely access your device

```
┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo ufw allow 22
Rule added
Rule added (v6)

┌──(kali㉿kali)-[~/ELI/day4]
└─$ _
```

Fig5

6. Now that we have denied and allowed important services lets try to connect to out device using ssh to check whether we successfully did it or not the full process is in fig 6

Home | task 4-3.pdf | Sign in | Upgrade now

Home Insert Edit Page Comment | Input searching co... | WPS AI | Share

PDF to Word | Export PDF to Picture | Play Slide | 101.16% | 1/2 | Rotate All Pages | One Page | Two Page

CYBER SECURITY INTERNSHIP

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

- **Objective: Configure and test basic firewall rules to allow or block traffic.**
- **Tools: Windows Firewall / UFW (Uncomplicated Firewall) on Linux.**
- **Deliverables: Screenshot/configuration file showing firewall rules applied.**

Hints/Mini Guide:

1. Open firewall configuration tool (Windows Firewall or terminal for UFW).
2. List current firewall rules.
3. Add a rule to block inbound traffic on a specific port (e.g., 23 for Telnet).
4. Test the rule by attempting to connect to that port locally or remotely.
5. Add rule to allow SSH (port 22) if on Linux.
6. Remove the test block rule to restore original state.
7. Document commands or GUI steps used.
8. Summarize how firewall filters traffic.

Outcome: Basic firewall management skills and understanding of network traffic filtering.

Interview Questions:

1. What is a firewall?
2. Difference between stateful and stateless firewall?
3. What are inbound and outbound rules?
4. How does UFW simplify firewall management?
5. Why block port 23 (Telnet)?
6. What are common firewall mistakes?
7. How does a firewall improve network security?
8. What is NAT in firewalls?

Key Concepts: Firewall configuration, network traffic filtering, ports, UFW, Windows Firewall.

Submit Here:
After completing the task, paste your GitHub repo link and submit it using the link below:
- [Submission Link]

1/2 | 101%

Ubuntu 64-bit Arm | 0:04

kali@kali: ~/ELI/day4

File Actions Edit View Help

```
udp6       0      0 fe80::7ded:a51c:7c:3702 :::*
udp6       0      0 ff02::c:3702            :::*
udp6       0      0 fe80::7ded:a51c:7c4:546 :::*

┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo systemctl start ssh

┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh
Created symlink '/etc/systemd/system/sshd.service' → '/usr/lib/systemd/system/ssh.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/ssh.service' → '/usr/lib/systemd/system/ssh.service'.

┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: disabled)
     Active: active (running) since Thu 2025-06-05 00:02:24 BST; 19s ago
 Invocation: 6516d18123f34ce88363cea2f549f6b1
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 60351 (sshd)
      Tasks: 1 (limit: 6299)
     Memory: 2M (peak: 2.8M)
        CPU: 20ms
     CGroup: /system.slice/ssh.service
             └─60351 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Jun 05 00:02:24 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Jun 05 00:02:24 kali sshd[60351]: Server listening on 0.0.0.0 port 22.
Jun 05 00:02:24 kali sshd[60351]: Server listening on :: port 22.
Jun 05 00:02:24 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.

┌──(kali㉿kali)-[~/ELI/day4]
└─$ _
```

chinmay — kali@kali: ~ — ssh kali@192.168.31.101 — 134×29

```
Last login: Thu Jun  5 02:40:28 on ttys000
chinmay@Chinmay-ka-Mini ~ % ssh kali@192.168.31.101
The authenticity of host '192.168.31.101 (192.168.31.101)' can't be established.
ED25519 key fingerprint is SHA256:RXT1JNnYe1BSuNObK5MRIKpfa1OEewEwMYih3D9UXYE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Host key verification failed.
chinmay@Chinmay-ka-Mini ~ % ssh kali@192.168.31.101
The authenticity of host '192.168.31.101 (192.168.31.101)' can't be established.
ED25519 key fingerprint is SHA256:RXT1JNnYe1BSuNObK5MRIKpfa1OEewEwMYih3D9UXYE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.31.101' (ED25519) to the list of known hosts.
kali@192.168.31.101's password:
Linux kali 6.12.20-arm64 #1 SMP Kali 6.12.20-1kali1 (2025-03-26) aarch64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
┌──(kali㉿kali)-[~]
└─$ ls
Desktop  Downloads  ELI  htb  rev  snap  thm  tools

┌──(kali㉿kali)-[~]
└─$
```

The breakdown of the whole process is:

- First step is to start ssh service and enable using to commands:
  " sudo systemctl start ssh " & " sudo systemctl enable ssh "
- Then check whether its runnnig or not using " sudo systemctl status ssh "
- Now that we know our ssh is accespting incoming connections let's try to connect to it using out mac terminal
- Simple ssh coonection request command is:
  " ssh username@IP "
- As you can see in fig6 we successfully established a connection with our linux machine and are able to access the file system.

Now let's remove our test block to restore the original state of the firewall :

1. The simple command to see what we just changed on our firewall is:
   " sudo ufw status numbered ".fig7

```
┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 23                         DENY IN     Anywhere
[ 2] 22                         ALLOW IN    Anywhere
[ 3] 23 (v6)                    DENY IN     Anywhere (v6)
[ 4] 22 (v6)                    ALLOW IN    Anywhere (v6)
```

Fig7

2. Lets remove the rule we added for telnet:fig8

```
┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo ufw delete 1
Deleting:
 deny 23
Proceed with operation (y|n)? y
Rule deleted

┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo ufw delete 3
Deleting:
 allow 22
Proceed with operation (y|n)? y
Rule deleted (v6)

┌──(kali㉿kali)-[~/ELI/day4]
└─$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 22                         ALLOW IN    Anywhere
[ 2] 23 (v6)                    DENY IN     Anywhere (v6)
```

Fig8

3. As you can see, we removed the conditions we added to telnet service

By this procedure let's understand how Firewall actually works and what we did to our firewall: Firewalls are **first-line defenses** in cybersecurity. Even if a service is not running (like Telnet), blocking its port is **proactive security**. UFW makes this process easy for Linux users with simple commands to manage complex iptables rules behind the scenes.

What a firewall does is whenever a someone or something is trying to get access to something on our device network it checks for security rules set by the OS or device owner if set manually and only allows it to access it if it matches the security ruleset conditions.