**Task 4: Setup and Use a Firewall on Linux (Kali – UFW)**

**Objective**

Configure and test basic firewall rules to allow or block traffic on a Linux system, specifically blocking Telnet (port 23) and ensuring SSH (port 22) remains accessible.

**Tools Used**

* **UFW** – Uncomplicated Firewall (CLI tool for managing iptables/nftables)
* **netcat-openbsd** – for testing open/blocked ports
* **Kali Linux** – OS used for demonstration

**Step-by-Step Process**

**1. Open Firewall Configuration Tool (UFW Terminal)**

**Install UFW:**

sudo apt update

sudo apt install ufw -y

*(Optional GUI)*

sudo apt install gufw -y

**Check firewall status:**

sudo ufw status verbose

Output (initially inactive):

Status: inactive

**2. List Current Firewall Rules**

Before any configuration:

sudo ufw status numbered

**3. Add a Rule to Block Inbound Traffic on Port 23 (Telnet)**

**Allow SSH first (to prevent lockout if remote):**

sudo ufw allow OpenSSH

or:

sudo ufw allow 22/tcp

**Enable firewall:**

sudo ufw enable

Confirm:

Command may disrupt existing ssh connections. Proceed with operation (y|n)? y

Firewall is active and enabled on system startup

**Block Telnet (port 23) traffic:**

sudo ufw deny 23/tcp

**Verify rule is added:**

sudo ufw status numbered

Example output:

Status: active

To Action From

-- ------ ----

22/tcp ALLOW Anywhere

23/tcp DENY Anywhere

**4. Test the Rule by Attempting to Connect to Port 23**

**Install Netcat for testing:**

sudo apt install netcat-openbsd -y

**Open a listener on port 23 in one terminal:**

sudo nc -l -p 23

*(Leave this running; simulates a Telnet service)*

**In another terminal, test connection:**

nc -vz localhost 23

* Before blocking:  
  Connection to localhost 23 port [tcp/telnet] succeeded!
* After blocking with UFW:  
  nc: connect to localhost port 23 (tcp) failed: Connection refused  
  *(or timeout if DROP policy is used)*

**5. Add Rule to Allow SSH (Port 22) if on Linux**

If not already added in Step 3:

sudo ufw allow 22/tcp

Verify:

sudo ufw status numbered

**6. Remove the Test Block Rule to Restore Original State**

sudo ufw delete deny 23/tcp

Confirm removal:

sudo ufw status numbered

**7. Document Commands Used**

Final command list for reference:

sudo apt update

sudo apt install ufw gufw netcat-openbsd -y

sudo ufw status verbose

sudo ufw status numbered

sudo ufw allow OpenSSH

sudo ufw enable

sudo ufw deny 23/tcp

sudo ufw status numbered

sudo nc -l -p 23

nc -vz localhost 23

sudo ufw delete deny 23/tcp

sudo ufw status numbered

**8. Summary – How Firewall Filters Traffic**

A firewall inspects network packets and applies rules to decide whether to **ALLOW**, **DENY**, or **DROP** them.

* **ALLOW** – Permits traffic to pass.
* **DENY / REJECT** – Blocks traffic (REJECT sends error back; DENY silently drops).
* **DROP** – Ignores traffic with no reply.
* **Filtering criteria** – Based on protocol (TCP/UDP), port, source/destination IP, and interface.
* **Stateful inspection** – Tracks existing connections and automatically allows related traffic.
* **Default policy** – If no rule matches, traffic is handled according to the firewall’s default policy (often deny incoming, allow outgoing).
* **UFW** – Provides a simple interface to Linux’s iptables/nftables for easy firewall rule management.

**Deliverables**

* **Tool Used:** UFW on Kali Linux
* **Rule Applied:** Deny inbound TCP traffic on port 23, allow SSH (port 22)
* **Testing:** Verified using netcat
* **Final State:** Restored firewall to initial configuration after testing.