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

Objective

The aim of this task was to gain hands-on experience in configuring and managing basic firewall rules to control network traffic. The exercise involved working with both UFW (Uncomplicated Firewall) on Linux and Windows Defender Firewall on Windows to allow or block specific traffic, understand firewall concepts, and test and validate firewall rules safely.

Tools Used

- UFW (Linux) – A simplified command-line interface for iptables, making firewall management user-friendly.
- Windows Defender Firewall (Windows) – A built-in GUI and command-line-based firewall with fine-grained inbound/outbound control.

Step-by-Step Implementation

On Linux (UFW)

1. Enabled UFW to ensure the firewall was active: sudo ufw enable
2. Listed existing rules for visibility: sudo ufw status numbered
3. Blocked port 23 (Telnet) to prevent insecure access: sudo ufw deny 23/tcp
4. Allowed SSH (port 22) for secure remote connections: sudo ufw allow 22/tcp
5. Tested rules using connection attempts and telnet/ssh commands.
6. Removed test rules to restore the default state: sudo ufw delete deny 23/tcp

On Windows (Defender Firewall)

1. Opened Windows Defender Firewall via Control Panel or Settings.
2. Navigated to Advanced Settings → Inbound Rules.
3. Created a new inbound rule to block TCP port 23 (Telnet).
4. Tested the rule using a Telnet client — connection was blocked.
5. Deleted the rule after verification to restore default configuration.

Extra Research Insights

- Integration with Other Security Tools – Firewalls work best when combined with IDS/IPS and endpoint protection.
- User-friendliness – UFW abstracts complex iptables syntax into readable commands, making it ideal for small-scale systems.
- Granular Control – Windows Defender Firewall allows filtering based on port, program, or protocol.

- Misconfiguration Risks – Blocking essential services can cause lockouts; leaving insecure ports open invites attacks.
- Stateful vs Stateless Firewalls – Stateful firewalls track session states for smarter decisions; stateless ones inspect packets individually.
- NAT and Security – NAT in firewalls masks internal IPs and allows multiple devices to share one public IP.

Key Concepts Covered

- Firewall Rules – Predefined conditions to allow or block network traffic.
- Inbound vs Outbound Traffic – Controlling external and internal connections.
- Port Blocking – Restricting access to specific network ports.
- Protocol Security – Avoiding insecure protocols like Telnet and using secure ones like SSH.
- Traffic Filtering – Controlling access based on source, destination, or protocol.
- Stateful vs Stateless Filtering – Context-aware vs packet-by-packet inspection.

Reflection Questions Explored

- What differentiates stateful from stateless firewalls?
- Why is Telnet (port 23) considered insecure?
- How does a firewall differ from antivirus software or IDS?
- What are the risks of misconfigured firewall rules?
- How does NAT enhance firewall security?

Learning Outcomes

By completing this task, I understood how firewalls regulate network traffic, learned how to apply rules via both GUI and CLI, practiced safe configuration and rollback, developed awareness of network security best practices, and understood the security benefits of replacing Telnet with SSH.