# Task 4: Setup and Use a Firewall (Windows & Linux)

## Objective:

The objective of this task is to configure and test basic firewall rules to enhance network security. Specifically, the task involves blocking the insecure Telnet service on port 23, while ensuring that secure remote access via SSH (port 22) remains allowed. The firewall is configured, tested, and restored to its original state on both Linux (using UFW) and Windows (using Windows Defender Firewall / PowerShell).

## Part A: Linux (UFW)

- Install and check UFW: `sudo apt update && sudo apt install ufw -y` followed by `sudo ufw status verbose`.

- List current firewall rules and save output for documentation.

- Add a blocking rule for Telnet: `sudo ufw deny 23/tcp`.

- Allow SSH to ensure administrative access: `sudo ufw allow 22/tcp`.

- Enable UFW (if not already enabled): `sudo ufw enable`.

- Test the rules by using tools like netcat (`nc -vz 23`) or nmap (`nmap -p 23 `). Expected result: port 23 blocked, port 22 open.

- Turn on logging: `sudo ufw logging on` and verify dropped packets in `/var/log/ufw.log`.

- Remove the blocking rule: `sudo ufw delete deny 23/tcp`.

- Save final firewall status and rules for evidence.

## Part B: Windows Firewall

- Open Windows Defender Firewall with Advanced Security (`wf.msc`).

- Navigate to Inbound Rules > New Rule. Choose Port > TCP > Specific port 23 > Block the connection.

- Create another rule for SSH (port 22) and set it to Allow.

- Verify rules appear in the Inbound Rules list.

- Test rules using PowerShell: `Test-NetConnection -ComputerName -Port 23`. Expected: False when blocked.

- Alternatively, use PowerShell commands:

- - List rules: `Get-NetFirewallRule`

- - Add block rule: `New-NetFirewallRule -DisplayName 'Block-Telnet-Test' -Direction Inbound -Action Block -Protocol TCP -LocalPort 23`

- - Allow SSH: `New-NetFirewallRule -DisplayName 'Allow-SSH' -Direction Inbound -Action Allow -Protocol TCP -LocalPort 22`

- - Remove rule: `Remove-NetFirewallRule -DisplayName 'Block-Telnet-Test'`.

- Export rules for documentation: `netsh advfirewall firewall show rule name=all > firewall_rules.txt`.

## Part C: Testing the Firewall Rules

Testing was conducted both locally (same machine) and remotely (from another host on the LAN). The following methods were used:

- Local listener using netcat (`nc -l 23`) to check blocked Telnet connections.

- Remote machine attempted connection: `nc -vz 23` (expected failure).

- Nmap scan: `nmap -p 23 ` (expected result: filtered or closed).

- Windows Test-NetConnection for port 23 returned False when blocked and True when allowed.

## Part D: Documentation and Deliverables

The following evidence should be included in the submission:

- Screenshots of firewall rules before and after changes (Linux UFW status, Windows inbound rule screenshot).

- Command outputs (saved into text files for both Linux and Windows).

- Final restored state showing firewall rules after temporary changes were removed.

- README file with summary of steps performed.

- Organized repository structure with folders for commands, outputs, and screenshots.

## Conclusion:

This task successfully demonstrated how to configure, test, and restore firewall rules on Linux and Windows. By blocking Telnet (port 23) and allowing SSH (port 22), the principle of least privilege was enforced, reducing the attack surface while maintaining administrative access. Logs, outputs, and screenshots provide verifiable proof of execution.