Use a Firewall on Windows/Linux

1. Open Firewall Configuration Tool

- **Windows**:

  - Open **Windows Defender Firewall with Advanced Security**:

    - Press `Win + R`, type `wf.msc`, and press Enter.

    - Alternatively, go to Control Panel → System and Security → Windows Defender Firewall → Advanced Settings.

- **Linux (UFW)**:

  - Open a terminal (e.g., Ctrl + T or use a terminal emulator).

  - Ensure UFW is installed: `sudo apt install ufw` (Debian/Ubuntu-based systems).

  - Verify UFW status: `sudo ufw status`.

Note: Administrative/root privileges are required for both.

---

2. List Current Firewall Rules

- **Windows**:

  - In **Windows Defender Firewall with Advanced Security**, select **Inbound Rules** or **Outbound Rules** from the left pane.

  - View the list of rules, including details like port, protocol, and action (allow/block).

  - For CLI, use PowerShell: `Get-NetFirewallRule | Format-Table Name,DisplayName,Enabled,Direction,Action`.

- **Linux (UFW)**:

  - Run `sudo ufw status verbose` to list active rules, showing allowed/blocked ports, protocols, and IP addresses.

  - Example output:

    ```
    Status: active
    To                  Action      From
    --                  ------      ----
    22/tcp              ALLOW       Anywhere
    80/tcp              DENY        Anywhere
    ```

---

3. Add a Rule to Block Inbound Traffic on a Specific Port (e.g., Port 23 for Telnet)

- **Windows**:

  - In **Windows Defender Firewall with Advanced Security**:

    1. Click **Inbound Rules** → **New Rule**.

    2. Select **Port** → Next.

    3. Choose **TCP**, enter `23` in **Specific local ports** → Next.

    4. Select **Block the connection** → Next.

    5. Apply to all profiles (Domain, Private, Public) → Next.

    6. Name the rule (e.g., “Block Telnet Port 23”) → Finish.

  - CLI (PowerShell, run as Administrator):

 ```powershell
    New-NetFirewallRule -Name "Block_Telnet_23" -DisplayName "Block Telnet Port 23" -Direction Inbound -Protocol TCP -LocalPort 23 -Action Block
    ```

- **Linux (UFW)**:
  - Run: `sudo ufw deny 23/tcp`
  - This blocks inbound TCP traffic on port 23 (Telnet).

---

4. Test the Rule by Attempting to Connect to the Port
- **Locally**:
  - Use a tool like **netcat** (`nc`) or **telnet**.
    - On Linux/macOS: `telnet localhost 23` or `nc -zv localhost 23`.
    - On Windows: `telnet 127.0.0.1 23` (if Telnet client is enabled).
    - Expected result: Connection refused or timeout, confirming the port is blocked.
- **Remotely** (from another device on the same network, with permission):
  - Identify the target machine’s IP (e.g., `192.168.1.100`).
  - Run: `telnet 192.168.1.100 23` or `nc -zv 192.168.1.100 23`.
  - Expected result: Connection refused or timeout.
- **Alternative**: Use `nmap` to scan: `nmap -p 23 <target_ip>`. A “closed” or “filtered” state confirms the block.

Note: Ensure no Telnet service is running on port 23, as it’s insecure and typically disabled by default.

---

5. Add Rule to Allow SSH (Port 22) if on Linux
- **Linux (UFW)**:
  - Run: `sudo ufw allow 22/tcp`

  - This allows inbound TCP traffic on port 22 (SSH).

  - Verify: `sudo ufw status` (should show `22/tcp ALLOW Anywhere`).

- **Windows** (if applicable, e.g., running an SSH server like OpenSSH):

  - In **Windows Defender Firewall with Advanced Security**:

    1. Click **Inbound Rules** → **New Rule**.

    2. Select **Port** → Next.

    3. Choose **TCP**, enter `22` in **Specific local ports** → Next.

    4. Select **Allow the connection** → Next.

    5. Apply to all profiles → Next.

    6. Name the rule (e.g., “Allow SSH Port 22”) → Finish.

  - CLI (PowerShell):

    ```powershell
    New-NetFirewallRule -Name "Allow_SSH_22" -DisplayName "Allow SSH Port 22" -Direction Inbound -Protocol TCP -LocalPort 22 -Action Allow
    ```

---

6. Remove the Test Block Rule to Restore Original State

- **Windows**:

  - In **Windows Defender Firewall with Advanced Security**:

    1. Select **Inbound Rules**.

    2. Find “Block Telnet Port 23”, right-click → **Delete**.

  - CLI (PowerShell):

    ```powershell
    Remove-NetFirewallRule -Name "Block_Telnet_23"
    ```

```

- **Linux (UFW)**:

  - Run: `sudo ufw delete deny 23/tcp`

  - Verify: `sudo ufw status` (port 23 rule should be gone).

---

### 7. Document Commands or GUI Steps Used

Below is a consolidated list of commands and GUI steps used:

- **Windows**:

  - Open Firewall: `wf.msc` or Control Panel → Windows Defender Firewall → Advanced Settings.

  - List Rules (CLI): `Get-NetFirewallRule | Format-Table Name,DisplayName,Enabled,Direction,Action`.

  - Block Port 23 (GUI): New Rule → Port → TCP → 23 → Block → All profiles → Name: "Block Telnet Port 23".

  - Block Port 23 (CLI): `New-NetFirewallRule -Name "Block_Telnet_23" -DisplayName "Block Telnet Port 23" -Direction Inbound -Protocol TCP -LocalPort 23 -Action Block`.

  - Allow Port 22 (GUI): New Rule → Port → TCP → 22 → Allow → All profiles → Name: "Allow SSH Port 22".

  - Allow Port 22 (CLI): `New-NetFirewallRule -Name "Allow_SSH_22" -DisplayName "Allow SSH Port 22" -Direction Inbound -Protocol TCP -LocalPort 22 -Action Allow`.

  - Remove Rule (GUI): Inbound Rules → Find "Block Telnet Port 23" → Delete.

  - Remove Rule (CLI): `Remove-NetFirewallRule -Name "Block_Telnet_23"`.

- **Linux (UFW)**:

  - Open UFW: Terminal, check status with `sudo ufw status`.

  - List Rules: `sudo ufw status verbose`.

  - Block Port 23: `sudo ufw deny 23/tcp`.
```

  - Allow Port 22: `sudo ufw allow 22/tcp`.

  - Remove Rule: `sudo ufw delete deny 23/tcp`.

---

8. Summarize How Firewall Filters Traffic

A firewall filters network traffic by enforcing rules that control which packets are allowed or blocked based on criteria like:

- **Source/Destination IP**: Specifies which devices can send/receive traffic.

- **Port Number**: Determines which services (e.g., port 23 for Telnet, 22 for SSH) are accessible.

- **Protocol**: Filters by protocol type (e.g., TCP, UDP).

- **Direction**: Manages inbound (incoming) or outbound (outgoing) traffic.

- **Action**: Allows, blocks, or redirects traffic.

How It Works:

- The firewall inspects packet headers against its rule set.

- Rules are processed in order (or priority). The first matching rule determines the action (allow/block).

- If no rule matches, the default policy (e.g., deny all) applies.

- Example: Blocking port 23 prevents Telnet connections, while allowing port 22 enables SSH access.

Outcome: These tasks demonstrate basic firewall management skills, including rule creation, testing, and documentation, and provide an understanding of how firewalls secure networks by filtering traffic.