Unit III
Network Defence Tools

---

1. Firewall: Introduction

i. A **firewall** is a network security device or software application that monitors and controls incoming and outgoing network traffic based on predefined security rules.
ii. Firewalls are essential for protecting computer networks from **unauthorised access, cyberattacks, malware, and other security threats**.
iii. A firewall analyses network traffic and makes decisions to **allow or block data packets** according to the configured security policies.
iv. It controls both **incoming (inbound)** and **outgoing (outbound)** traffic to ensure that only legitimate and authorised communication is permitted.
v. Firewall rules can be defined based on parameters such as **IP addresses, port numbers, protocols, application types, and traffic direction**.
vi. The table below illustrates a **typical firewall rule table** used to manage and filter network traffic.

Example: Firewall Rule Table

Rule No.	Source IP	Destination IP	Port No.	Protocol	Action
1	192.168.1.10	Any	80	HTTP	Allow
2	Any	192.168.1.20	22	SSH	Block
3	192.168.1.0/24	Any	443	HTTPS	Allow

4	Any	Any	Any	ICMP	Block
5	Any	Any	Any	Any	Deny

- The firewall checks traffic **rule by rule**, usually from top to bottom.
- Once a matching rule is found, the corresponding **action (Allow/Block/Deny)** is applied.
- The final rule often acts as a **default deny rule** to block unspecified traffic.

vii. A **firewall** can be implemented as either a **hardware device**, a **software application**, or a combination of both, depending on the network requirements.

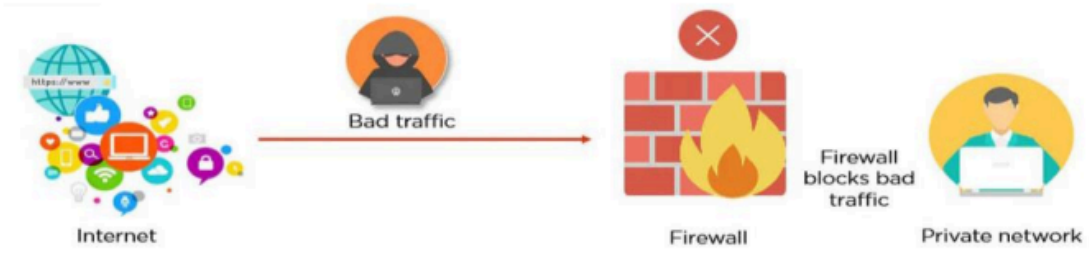
viii. Firewalls primarily operate at the **Network Layer (Layer 3)** and **Transport Layer (Layer 4)** of the OSI model to filter traffic based on IP addresses, ports, and protocols.

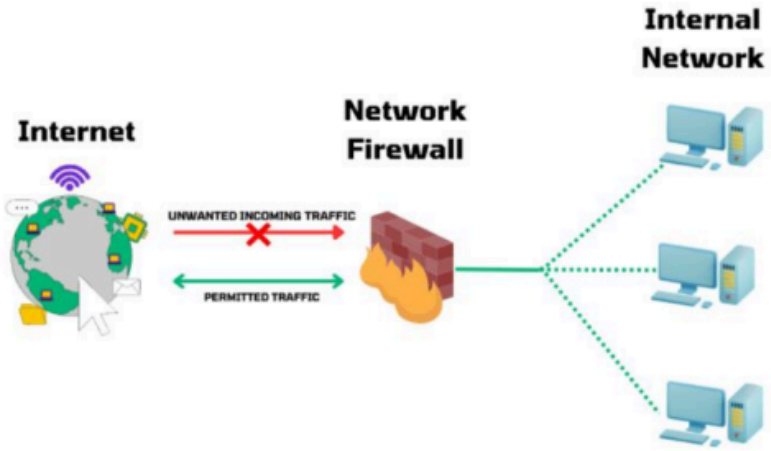
ix. A firewall can **block or allow specific IP addresses and network services**, such as HTTP, HTTPS, FTP, SSH, and other commonly used protocols.

x. **Access Control:** Firewalls enforce access control policies by determining which **users, systems, or devices** are permitted to access specific network resources. This helps prevent **unauthorised access and data breaches**.

xi. **Application Control:** Advanced or next-generation firewalls can control access to **specific applications and services**, enforcing security policies based on the application itself rather than only on IP addresses and port numbers.

xii. Firewalls are a **critical component of a comprehensive network security strategy**, providing a **first line of defence** against cyber threats and helping to ensure the **confidentiality, integrity, and availability (CIA triad)** of network resources.





Types of Firewalls

1) Packet-Filtering Firewalls

Packet-filtering firewalls are the **most basic type of firewall**. They examine individual data packets and decide whether to allow or block them based on predefined rules.

- Decisions are made based on:
    - Source IP address
    - Destination IP address
    - Port number
    - Protocol (TCP, UDP, ICMP, etc.)
- They do not inspect the content of packets.
- They are fast but provide **limited security**.

Example: Blocking all incoming traffic on port 23 (Telnet).

---

2) Stateful Inspection Firewalls

Stateful inspection firewalls keep track of the **state of active network connections**.

- They monitor ongoing sessions and understand whether a packet is part of an existing, legitimate connection.
- More secure than packet-filtering firewalls.
- Widely used in modern networks.

Example: Allowing incoming traffic only if it is a response to an outgoing request.

---

3) Proxy Firewalls

Proxy firewalls act as an **intermediary** between users and the internet.

- They receive requests from clients and forward them to the destination server.
- They hide internal IP addresses and provide strong security.
- They can filter content and applications.

Disadvantage: Slower performance due to additional processing.

---

4) Next-Generation Firewalls (NGFWs)

Next-Generation Firewalls provide **advanced security features** beyond traditional firewalls.

- Features include:
    - Application awareness and control
    - Intrusion Prevention System (IPS)

    - Deep packet inspection
    - Malware protection
- Used in enterprise and large organisational networks.

1.1 Linux Firewall

A **Linux firewall** is a security tool that protects a Linux system or server from unauthorised access and cyberattacks. It works like a **gatekeeper**, controlling incoming and outgoing network traffic based on defined rules.

Key Concepts

1. Firewall Basics

A firewall works like a **security guard at a building entrance**. Each data packet is checked, and only those that follow the security rules are allowed.

---

2. Packet Filtering

Linux firewalls filter **data packets**, which are small units of data transmitted over a network. Rules can be based on:

- Source address
- Destination address
- Protocol (HTTP, HTTPS, SSH, etc.)

---

3. iptables

`iptables` is the most commonly used firewall tool in Linux.

- **Chains:**

    - INPUT – Incoming traffic
    - OUTPUT – Outgoing traffic
    - FORWARD – Traffic passing through the system
- **Tables:**
    - Filter table – Used for basic packet filtering

---

4. Allowing or Blocking Traffic

Examples of firewall rules:

- Allow traffic from a trusted IP address
- Block traffic on a specific port
- Allow only selected services such as HTTP or SSH

---

5. Default Policies

Default policies decide what happens when no rule matches.

- Common default: **Deny all traffic unless explicitly allowed**

Simple iptables Commands

- **Allow HTTP traffic (Port 80):**

```
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

- **Block all incoming traffic:**

```
sudo iptables -P INPUT DROP
```

Why Use a Linux Firewall?

- **Security:** Protects against hackers and attacks
- **Control:** Manages which services can access the network
- **Privacy:** Prevents unauthorised data access

Real-Life Example

If you host a website on a Linux server, you can allow only your organisation's IP addresses to access the administration panel, blocking all other connections.

1.2 Windows Firewall

Windows Firewall is a built-in security feature in Microsoft Windows that monitors and controls incoming and outgoing network traffic.

Why Should You Use Windows Firewall?

1. **Protection Against Hackers and Malware**
    - Blocks malicious connections and unauthorised access.
2. **Prevents Unauthorised Access**
    - Stops attackers from exploiting open ports.
3. **Monitors Network Traffic**
    - Allows only trusted applications to communicate.
4. **Customisable Security Settings**
    - Users can allow or block specific programs.
5. **Prevents Data Theft**
    - Blocks suspicious outgoing connections.

How Windows Firewall Works

- **Incoming Traffic:** Blocks traffic from untrusted sources.
- **Outgoing Traffic:** Prevents unauthorised programs from sending data.

Examples of Windows Firewall in Action

- Blocking internet access for unknown applications
- Preventing hackers from accessing shared resources
- Allowing trusted programs like browsers and games
- Stopping malware from sending stolen data

Should You Turn Off Windows Firewall?

It is **not recommended** to turn off Windows Firewall, as it exposes the system to security threats. It may be disabled temporarily only for troubleshooting.

Key Differences Between Linux and Windows Firewalls

Feature	Linux Firewall	Windows Firewall
Customisation	Highly customisable	User-friendly
Interface	Command-line	GUI + Command-line
Use Case	Servers, advanced users	Home and office PCs
Configuration	Requires networking knowledge	Easy to configure
Default Behaviour	Often restrictive	Generally permissive

Conclusion

Both **Linux and Windows firewalls** serve the same fundamental purpose of protecting systems from unauthorised access.

- **Linux firewalls** provide advanced control and flexibility, making them ideal for servers.
- **Windows firewalls** are easier to use and suitable for everyday users.

Firewalls are a **crucial component of network security**, ensuring that only authorised and safe traffic is allowed within a system.

2. Packet Filters & Packet Filter vs Firewall

What is a Packet Filter?

A **packet filter** is a basic network security mechanism used to control the flow of data packets between different networks, such as the internet and a local network. It operates at the **Network Layer (Layer 3)** of the OSI model.

A packet filter examines the **header information** of each incoming and outgoing packet, including:

- Source IP address
- Destination IP address
- Source port number
- Destination port number
- Protocol type (TCP, UDP, ICMP, etc.)

Based on predefined rules, the packet filter **allows or blocks** packets. Packet filters are mainly used to enforce **simple access control policies**, such as permitting or denying traffic from specific IP addresses or port numbers.

Aspect	Packet Filter	Firewall
Function	Filters packets based on header information (IP, ports, etc.)	Provides more comprehensive protection, including packet filtering and other security measures like stateful inspection.
Layer of Operation	Operates at the **Network Layer (Layer 3)** of the OSI model.	Operates at multiple layers, including the **Network Layer** and **Application Layer (Layer 7)**.
Statefulness	**Stateless**: Evaluates each packet independently.	**Stateful**: Tracks the state of connections and evaluates packets in the context of that state.
Complexity	Simpler and faster but less flexible.	More complex and slower but more versatile and secure.
Security Features	Basic filtering based on IP addresses, ports, and protocols.	Includes deep packet inspection, application filtering, VPN support, and more.
Use Case	Best for simple networks or low-security needs.	Suitable for enterprise networks requiring robust security.

Summary

- A **packet filter** is a basic form of network security that allows or denies traffic based on packet header information.

- A **firewall** is a more advanced security system that includes packet filtering along with **stateful inspection, deep packet inspection, and application control**, providing stronger protection against network threats.

3. Stateless vs Stateful Firewalls

Stateful Firewall

A **stateful firewall** keeps track of the **state of active network connections**, similar to maintaining a logbook of conversations. It does not examine packets individually in isolation; instead, it understands whether a packet is part of an existing and legitimate connection.

If a system inside the network initiates a connection, the firewall remembers it and allows the corresponding response traffic.

Example:

When you request a webpage, the firewall recognises that the request was initiated from inside the network and allows the server's response to return. Any unsolicited or unexpected packets are blocked.

Stateless Firewall

A **stateless firewall** does not maintain any information about ongoing connections. It treats each packet independently and checks it only against predefined rules.

It does not know whether a packet is part of a valid session or an attack.

Example:

If a response packet arrives at your computer, the stateless firewall checks only the packet details (IP, port, protocol). Even a legitimate response may be blocked if it does not match the rules.

Comparison: Stateless vs Stateful Firewalls

Feature	Stateless Firewall	Stateful Firewall
Connection Tracking	No	Yes
Packet Analysis	Individual packets only	Packets with session context
Security Level	Lower	Higher
Complexity	Simple	More complex
Performance	Faster	Slightly slower
Use Case	Simple networks	Enterprise and modern networks

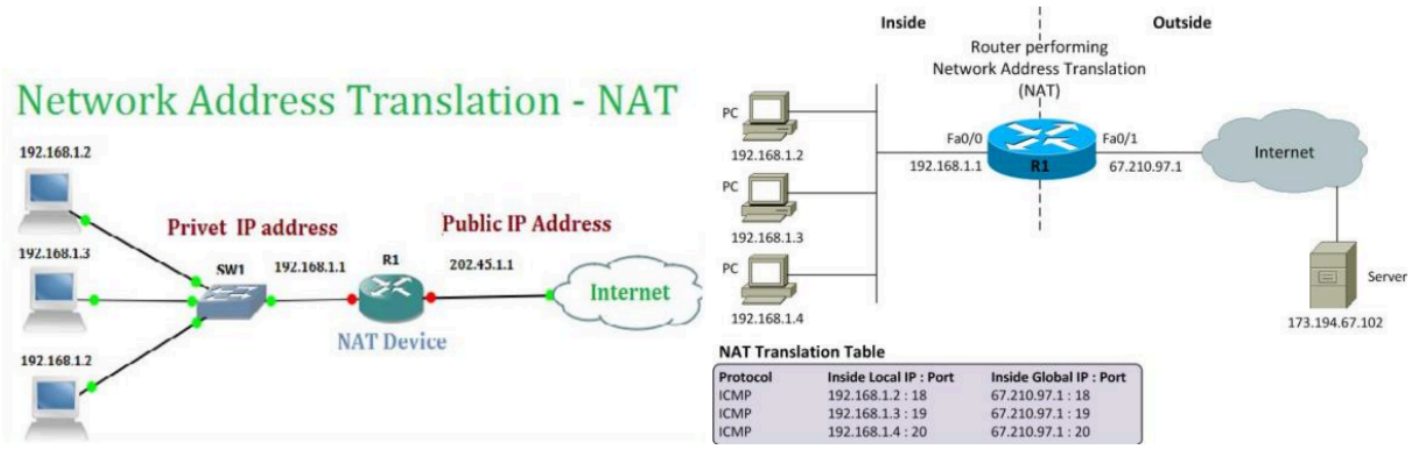
Summary

- **Stateful firewalls** track ongoing connections and provide better security by understanding traffic context.
- **Stateless firewalls** analyse packets individually, making them faster but less secure.
- In practice, **stateful firewalls are preferred** for most modern networks due to their improved protection capabilities.

4.1 Network Address Translation (NAT)

Network Address Translation (NAT) is a networking technique used by **routers or firewalls** to modify the **IP address information** in data packets as they pass through a network. NAT acts like a **translator**, allowing devices with private IP addresses to communicate with external networks such as the internet.

When a device inside a local network sends data to the internet, the router using NAT **replaces the private IP address with a public IP address**. Similarly, when a response comes back, the router translates the public IP address back to the original private IP address and forwards it to the correct internal device.



Why NAT is Used

- **Conserves public IP addresses** by allowing multiple devices to share a single public IP.
- **Improves security** by hiding internal (private) IP addresses from external networks.
- Enables internal devices to access the internet even though private IP addresses are **not routable** on the public internet.

How NAT Works (Simple Explanation)

1. A device inside the local network has a **private IP address** (e.g., 192.168.1.10).
2. The device sends a request to the internet.
3. The router replaces the private IP with its **public IP address**.
4. The response from the internet is sent to the router's public IP.
5. The router translates the address back and delivers the data to the correct internal device.

Types of NAT (Brief Overview)

- **Static NAT:** One private IP is mapped to one public IP.
- **Dynamic NAT:** Private IPs are mapped to public IPs from a pool.
- **PAT (Port Address Translation):** Multiple private IPs share one public IP using different port numbers (most common).

Real-Life Example

In a college or home network, many computers use private IP addresses. NAT allows all of them to access the internet using a **single public IP address** provided by the Internet Service Provider (ISP).

Summary

- NAT translates **private IP addresses into public IP addresses** and vice versa.
- It is commonly implemented on **routers and firewalls**.
- NAT helps in **IP address conservation, security, and efficient network communication**.

4.2 Port Forwarding

Port Forwarding is a networking technique used to allow **external devices or services** on the internet to access **specific services inside a private network**. It is commonly configured on a **router or firewall**.

Normally, routers block incoming connections from the internet to protect internal devices. Port forwarding overcomes this limitation by instructing the router to **forward traffic arriving on a specific port** to a designated device within the private network.

In simple terms, port forwarding works like **opening a gate** for a particular type of communication while keeping the rest of the network secure.

How Port Forwarding Works

1. A device inside the network has a **private IP address** (e.g., 192.168.1.10).
2. The router has a **public IP address** provided by the ISP.
3. The router is configured with a port forwarding rule.
4. When traffic arrives at the router on a specific port, it is **redirected to the internal device**.
5. Other unsolicited traffic continues to be blocked for security.

Simple Example

- A computer inside a home network runs a **game server**.
- The computer's private IP address is **192.168.1.10**.
- A port forwarding rule is set on the router:
    - *Public Port:* 12345
    - *Private IP:* 192.168.1.10
- When a friend connects to the router's public IP on port 12345, the router forwards the request to the game server inside the network.

As a result, external users can access the service while the rest of the network remains protected.

Common Uses of Port Forwarding

- Online gaming servers
- Web servers (HTTP/HTTPS)
- Remote desktop access
- CCTV and IP cameras
- File and media servers

Security Considerations

- Only forward **necessary ports**.
- Use strong authentication on forwarded services.
- Avoid exposing sensitive services directly to the internet.
- Combine port forwarding with firewall rules for better security.

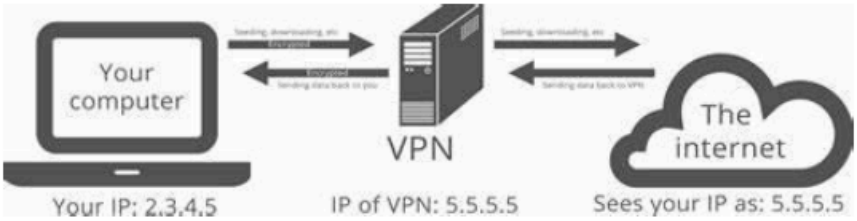
Summary

- Port forwarding allows **external access to internal network services**.
- It works by forwarding traffic from a **specific port on a router** to a **private IP address**.
- It is widely used for applications that require access from outside a private network.
- Improper configuration can lead to **security risks**, so it must be used carefully.

5. Virtual Private Network (VPN)

A **Virtual Private Network (VPN)** is a technology that provides a **secure and private connection** over the public internet. It protects users' online activities by creating an **encrypted tunnel** between the user's device and a VPN server. This ensures that data remains safe even when using **public Wi-Fi networks**.

When a VPN is enabled, all internet traffic is routed through the VPN server before reaching its destination. As a result, the user's **original public IP address is hidden**, and a **new IP address assigned by the VPN server** is used instead.



How a VPN Works (Simple Breakdown)

1. Security and Privacy

When a VPN is used, the internet connection is **encrypted**. Encryption converts readable data into an unreadable format, ensuring that sensitive information such as **passwords, messages, and browsing activity** cannot be easily accessed by hackers, websites, or even Internet Service Providers (ISPs).

2. Changing Your Location

A VPN allows users to **appear as if they are browsing from a different geographical location**. For example, if a user is located in the United States and connects to a VPN server in the United Kingdom, websites will detect the connection as originating from the UK. This is achieved by routing internet traffic through the VPN server in the selected location.

3. Accessing Blocked Content

Certain websites and online services restrict access based on geographical location. A VPN helps **bypass such geographical restrictions** by masking the user's real location and assigning an IP address from another country. This enables users to access region-restricted content securely.

4. Public Wi-Fi Safety

Public Wi-Fi networks, such as those found in cafés, airports, and hotels, are often unsecured. Using a VPN on public Wi-Fi protects data by **preventing attackers from intercepting network traffic**, thereby ensuring secure communication.

5. How It Works

• Without a VPN: When you connect to the internet, your device communicates directly with websites or services. Your internet service provider (ISP) can see what websites you visit.

• With a VPN: Your device first connects to the VPN server. Then, the server connects to websites for you, making it harder for anyone (including your ISP) to track your activity. Your real IP address (which can reveal your location) is hidden, and websites only see the IP address of the VPN server.

In Summary

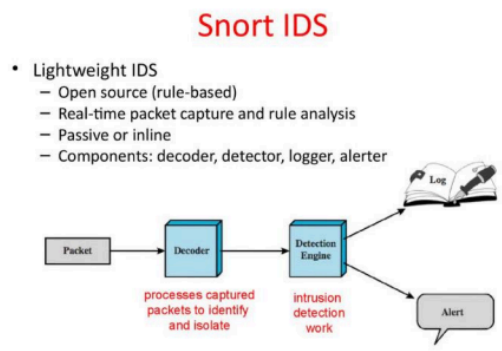
When you use a VPN (Virtual Private Network), your IP address changes to reflect the IP address of the VPN server you are connected to, rather than your own public IP address. The IP address seen by websites and online services is no longer your original IP address, but instead the IP address of the VPN server. This is because your internet traffic is being routed through the VPN server, so it appears to come from that server.

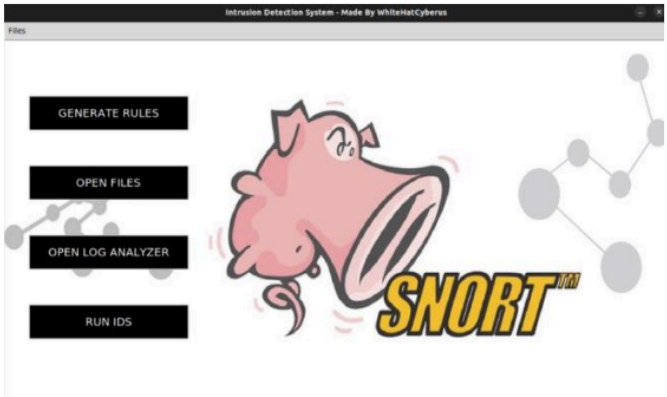
6. Snort: Intrusion Detection System (IDS)

Introduction to Intrusion Detection System (IDS)

Snort is an **open-source network security tool** used to detect and prevent unauthorised access or attacks on computer networks. It functions as an **Intrusion Detection System (IDS)** and can also operate as an **Intrusion Prevention System (IPS)** when configured accordingly.

Snort is widely used in organisations due to its **real-time traffic analysis**, flexibility, and strong community support.





What Does Snort Do?

The primary function of Snort is to **monitor network traffic** and detect suspicious or malicious activities such as:

- Hacking attempts
- Malware and virus activity
- Denial-of-Service (DoS) attacks
- Policy violations

When suspicious behaviour is detected, Snort generates **alerts** to inform system administrators.

How Does Snort Work?

Snort analyses **data packets** travelling across the network and identifies potential threats using the following detection techniques:

1. Signature-Based Detection

Snort compares network traffic against a **database of known attack signatures**.

- If a match is found, the traffic is flagged as malicious.
- Similar to how antivirus software detects known viruses.

2. Anomaly-Based Detection

Snort monitors normal network behaviour and looks for **unusual or abnormal patterns**.

- Any deviation from expected behaviour triggers an alert.
- Useful for detecting new or unknown attacks.

3. Protocol-Based Detection

Snort analyses **network protocols** such as HTTP, FTP, SMTP, and DNS.

- It checks whether protocols are being used incorrectly or exploited.
- Alerts are generated when protocol violations are detected.

Why is Snort Useful?

- **Security Monitoring:**
  Snort provides real-time monitoring of network traffic and helps protect networks by identifying and alerting administrators about potential security threats and cyberattacks.
- **Network Forensics:**
  Snort records detailed logs of network activity, which can be used to trace, analyse, and investigate security incidents after an attack has occurred.
- **Open-Source and Customisable:**
  Snort is an open-source tool, meaning it is free to use and can be modified according to specific security requirements. This makes it a popular choice for organisations, educational institutions, and individuals.

Key Features of Snort

- Open-source and free to use

- Real-time packet inspection
- Highly configurable rule-based engine
- Generates detailed alerts and logs
- Supports IDS and IPS modes

Real-Life Example

If an attacker attempts a **port scan** or tries to exploit a web server vulnerability, Snort can detect the pattern, generate an alert, and notify the network administrator.

Conclusion

In simple terms, Snort acts like a "watchdog" for computer networks. It continuously monitors network traffic, detects suspicious or malicious activities, and alerts users when potential threats are identified. Due to its effectiveness, flexibility, and open-source nature, Snort is widely used as a reliable Intrusion Detection System for enhancing network security.