# Intro to Firewalls

A concise, beginner-friendly overview of firewalls: what they are, how they evolved, the major types (packet-filtering, stateful, application/WAF, and NGFW), deployment models (hardware, software, cloud), and hands-on management basics for Windows and Linux.

**TL;DR:** A firewall monitors and controls network traffic using rules. Think of it like a building's fire wall: it prevents "flames" (malware) from spreading between "rooms" (networks/hosts).

---

## What Is a Firewall?

A **firewall** is a network security control that monitors and filters **incoming and outgoing** traffic to prevent unauthorized access and cyberattacks.

**Analogy:**

- **House** = Network
- **Fire** = Malware/threats
- **Firewall** = Barrier that limits the spread

---

## A (Very) Short History

- **1988:** Digital Equipment Corporation (DEC) ships the first **packet filter** firewall.
- **Late 1990s: Application-layer firewalls** appear for deeper inspection.
- **2009:** Gartner coins **Next-Generation Firewall (NGFW)** — adds application awareness, IPS, and malware detection.

Firewalls remain a **first line of defense** and should be enabled on endpoints and networks.

---

# Core Types of Firewalls

## 1) Packet-Filtering Firewall

Simple rule-based filtering on packet headers (no payload inspection).

Typical criteria: **source/destination IPs**, **ports**, **protocols**.

**Analogy:** Security at a private event checking the **guest list**.

- Visitors = traffic
- Security = firewall
- Guest list = rules/ACLs

**Pros:** Lightweight, fast, easy for small networks.

**Limitation:** Can't detect malicious **content** hidden inside allowed packets.

---

## 2) Stateful Firewall

Tracks **connection state** (e.g., new/established/closing) and uses context to allow **return** traffic automatically.

**History:** Introduced by **AT&T Bell Labs (1989)**.
**Analogy:** If a guest inside invites their partner, security verifies the relationship and lets them in—even if not on the original list.

**Example: Microsoft Defender Firewall (Windows 11)** is stateful.

**Key difference vs. packet filter:**

- **Packet filter**: Static, per-packet rules only.
- **Stateful**: Dynamic, uses **connection history/context**.

---

## 3) Application Firewall (Layer 7) / WAF

Operates at **OSI Layer 7**; inspects **headers + content** (deep packet inspection). Can enforce **per-application** policies.

**Protects against:** SQL injection, XSS, command injection, etc.

**Types:**

- **WAF** (HTTP/HTTPS for web apps)
- **Database firewall**
- **Email firewall** (spam/phishing)

---

## 4) Next-Generation Firewall (NGFW)

Adds **app awareness**, **malware detection**, and **intrusion prevention (IPS)** to traditional firewalling. Often integrates with broader security tooling.

**Great for:** Highly regulated or threat-targeted environments (finance, healthcare, etc.).

---

## Quick Comparison — App Firewall vs. NGFW

| Feature | Application Firewall | NGFW |
|---|---|---|
| **OSI Layer** | Layer 7 | Multiple (incl. L7) |
| **Focus** | Specific applications | Whole network & threats |
| **Deep Packet Inspection** | ✅ | ✅ (more advanced) |
| **Stops App Attacks** | ✅ | ✅ |
| **Malware Detection** | ❌ | ✅ |

| | | |
|---|---|---|
| **Intrusion Prevention** | ❌ | ✅ |
| **Ecosystem Integration** | Sometimes | Often |

---

# Hardware vs. Software Firewalls

## Hardware Firewall

Physical appliance protecting whole networks (perimeter/DMZ/segments).

**Best for:** Medium–large orgs; centralized control. *(Often licensed.)*

## Software (Host) Firewall

Installed on a specific device (e.g., **Windows Defender Firewall**).
**Best for:** Individual systems, small offices, or additional host-level control.

Includes **container** and **virtual/cloud** firewalls.

**Comparison**

| Feature | Hardware | Software |
|---|---|---|
| **Form** | Physical device | Application |
| **Scope** | Entire network | Single host |
| **Placement** | Perimeter/DMZ/segments | On the device |

| | | |
|---|---|---|
| **Best For** | Medium–large orgs | Individuals/small offices |
| **Examples** | Cisco ASA, Fortinet, Palo Alto | Windows Defender Firewall |

---

# Deployment Models

## 1) Network-Based Firewalls

- **Purpose:** Protect an entire network.
- **Placement:** Perimeter, DMZ, or between **internal segments**.
- **Functions:** Packet filtering, DPI, NAT.
- **Pros:** Broad protection. **Cons:** Complexity, cost/training.

## 2) Host-Based Firewalls

- **Purpose:** Protect a single device/VM.
- **Functions:** Often **application-aware**, GUI-friendly.
- **Pros:** Granular control. **Cons:** Per-host management (unless centralized via **Group Policy**, etc.).

## 3) Combined (Layered) Use

- **SOHO/home:** Router firewall + host firewalls may be enough.
- **Enterprises:** Network firewall at the perimeter **plus** host firewalls on critical systems.

**Summary**

| Feature | Network-Based | Host-Based |
|---|---|---|
| **Scope** | Entire network | Single device |

| | | |
|---|---|---|
| **Placement** | Perimeter/DMZ/segments | On device/VM |
| **Complexity** | High | Low–Medium |
| **Cost** | Higher | Lower |
| **Resources** | Separate hardware | Uses host CPU/RAM |

### 4) Cloud-Based Firewalls & FWaaS

- Lives in **Azure/AWS/GCP** or similar.
- Helpful for **remote work** and **cloud apps** (traffic doesn't need hair-pinning through on-prem).
- Managed centrally; can protect cloud resources and connected sites.
- **FWaaS:** Third-party manages deployment/updates for you.

---

# Key Components & Best Practices

1. **Use a firewall** (endpoint + network). Keep it **enabled**.
2. **Harden config:** No default creds; understand your architecture; back up rules.
3. **Monitor & log:** Review logs; enable logging on drops/accepts.
4. **Seek expertise** where needed (MSPs/MSSPs).
5. **Layered security:** Firewalls ≠ anti-malware. Use **AV/EDR**, **VPN**, **IDS/IPS**, and keep systems **patched**.

---

# Managing Firewalls: Windows Client

### Network Profiles

- **Private:** Trusted (home/office). Allows discovery/sharing.
- **Public:** Untrusted (cafés/airports). Stricter rules, hide device.
- **Domain:** Corporate AD domain.

**Tips:** Choose the correct profile; keep firewall **ON** for all profiles; consider **"Block all incoming"** on public networks.

## Allowing/Blocking Apps

- *Settings → Privacy & Security → Windows Security → Firewall & network protection → Allow an app through firewall.*
- Choose app + profile(s).

## Advanced Settings (WFAS)

- *Control Panel → Windows Defender Firewall → Advanced settings.*
- **Inbound/Outbound rules**, **Connection Security rules**, **Monitoring**.
- **Logging:** Enable logging for dropped/successful connections.
- **Stateful:** Return traffic for established connections is allowed automatically.

**Example — Allow TCP Port 80 (Outbound):**

1. Outbound Rules → **New Rule** → **Port**.
2. **TCP**, **Specific port**: `80`.
3. **Allow the connection** → select **Domain/Private/Public**.
4. Name it (e.g., "Allow TCP 80").

---

# Managing Firewalls: Linux (Ubuntu/Debian)

Linux uses **netfilter** in the kernel; common tools: **iptables**, **firewalld**, **UFW** (default Ubuntu), **GUFW** (GUI).

## Install (if needed)

```
sudo apt install ufw      # CLI

sudo apt install gufw     # GUI
```

### Status / Enable / Disable

```
sudo ufw status

sudo ufw enable

sudo ufw disable
```

### Defaults (choose policy)

```
sudo ufw default deny incoming

sudo ufw default allow outgoing

# or:

sudo ufw default reject incoming
```

### Open a Port / Service

```
sudo ufw allow 22/tcp          # SSH

sudo ufw allow 80,443/tcp      # HTTP/HTTPS

sudo ufw delete allow 22/tcp   # remove rule
```

**GUFW Notes:** Profiles (Home/Office/Public), logging toggle, and simple **Allow/Deny/Reject** actions.

---

# Glossary

- **ACL:** Access Control List — rules that permit/deny traffic.
- **DPI:** Deep Packet Inspection — looks beyond headers into payloads.
- **DMZ:** Demilitarized Zone — isolates public-facing services.
- **IPS/IDS:** Intrusion Prevention/Detection Systems.
- **NAT:** Network Address Translation — remaps IP addresses/ports.
- **State Table:** Tracks connections for stateful filtering.